ENERGY-REGULARIZED SEQUENTIAL MODEL EDITING ON HYPERSPHERES

Anonymous authors
Paper under double-blind review

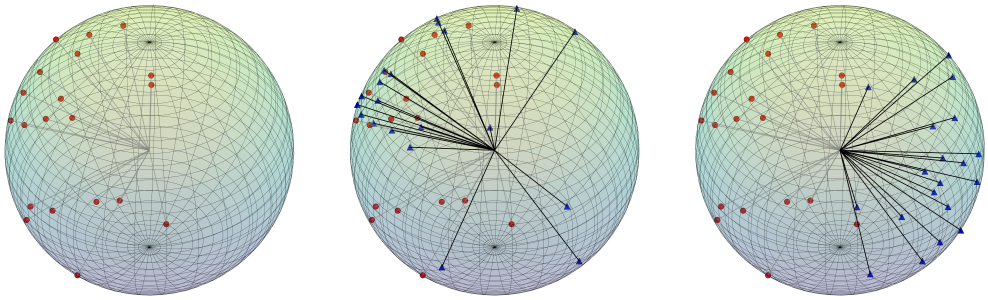
ABSTRACT

Large language models (LLMs) require constant updates to remain aligned with evolving real-world knowledge. Model editing offers a lightweight alternative to retraining, but sequential editing that updates the LLM knowledge through multiple successive edits often destabilizes representations and induces catastrophic forgetting. In this work, we seek to better understand and mitigate performance degradation caused by sequential editing. We hypothesize that *hyperspherical uniformity*, a property that maintains uniform distribution of neuron weights on a hypersphere, helps the model remain stable, retain prior knowledge, while still accommodate new updates. We use Hyperspherical Energy (HE) to quantify neuron uniformity during editing, and examine its correlation with editing performance. Empirical studies across widely used editing methods reveals a strong correlation between HE dynamics and editing performance, with editing failures consistently coinciding with uncontrolled HE fluctuations. We further theoretically prove that HE dynamics impose a lower bound on the degradation of pretrained knowledge, highlighting why HE stability is crucial for knowledge retention. Motivated by these insights, we propose SPHERE (**S**parse **P**rojection for **H**yperspherical **E**nergy-**R**egularized **E**diting), an HE-driven regularization strategy that stabilizes neuron weight distributions, ultimately preserving prior knowledge while enabling reliable sequential updates. Specifically, SPHERE identifies a sparse space complementary to the principal hyperspherical directions of the pretrained weight matrices and projects new knowledge onto it, attenuating perturbations on the principal directions. Extensive experiments on LLaMA3 (8B) and Qwen2.5 (7B) show that SPHERE outperforms the best baseline in editing capability by an average of 16.41%, while most faithfully preserving general model performance, thereby offering a principled path toward reliable large-scale knowledge editing.

1 INTRODUCTION

Large language models (LLMs) have demonstrated strong capabilities in knowledge storage, reasoning, and generation (DeepSeek-AI et al., 2024; Meta AI, 2024; Yang et al., 2025; OpenAI, 2025). However, the knowledge embedded in LLMs inevitably becomes outdated or incorrect, as real-world facts continuously evolve (Ji et al., 2023; Huang et al., 2025). Retraining LLMs to incorporate such updates is prohibitively expensive, motivating the development of *model editing* (also known as *knowledge editing*) (Cao et al., 2021; Mitchell et al., 2022; Meng et al., 2023; Gu et al., 2024; Fang et al., 2025). The most practical setting for model editing is *sequential editing*, where multiple updates are applied over time. However, previous studies have shown that such interventions often suffer from significant performance degradation due to catastrophic forgetting (Gu et al., 2024; Gupta et al., 2024). Consequently, reconciling the trade-off between preserving original pretrained knowledge and integrating new editing knowledge remains an unresolved challenge.

In this work, we seek to better understand and mitigate the performance degradation caused by sequential editing. We revisit model editing from the perspective of *hyperspherical uniformity* of perturbed weights (Liu et al., 2021), motivated by the observation that sequential edits often disrupt weight geometry, leading to degraded representations. Previous studies have shown that viewing a weight matrix as a set of neurons on a hypersphere (as shown in Figure 1 (a)) and maintaining their hyperspherical uniformity is crucial for stable training and effective generalization (Cogswell et al., 2016; Xie et al., 2017a; Qiu et al., 2023). To investigate the applicability of these principles



(a) Weight Neurons (b) Current Methods (c) SPHERE (Ours)

Figure 1: (a) A weight matrix is viewed as a set of neurons (red dots) on a hypersphere. (b) Current SOTA methods (Ma et al., 2025; Fang et al., 2025) introduce perturbations (blue triangles) that interfere with the principle hyperspherical directions of pre-edit weights. (c) SPHERE projects new knowledge onto a sparse space complementary to the principal hyperspherical directions.

to sequential editing, we adopt *hyperspherical energy (HE)* (Liu et al., 2018; Qiu et al., 2023) as a measure to quantify weight uniformity throughout sequential editing. HE calculates the dispersion of neuron weight vectors on a hypersphere, where lower energy corresponds to a more balanced distribution of neurons. By tracking HE dynamics throughout sequential editing, we can better understand how edits affect weight uniformity, identify early signs of destabilization, and even develop HE-driven regularization strategies to stabilize the editing process.

To reveal the mechanisms underlying successful editing strategies from the perspective of hyperspherical uniformity, we first empirically analyze how HE evolves throughout sequential editing and examine how these dynamics relate to editing performance across six widely used methods. Experimental results reveal a strong correlation between hyperspherical uniformity and editing performance, with editing failures consistently coinciding with its collapse. Meanwhile, more advanced editing methods have proven more effective at preserving hyperspherical uniformity. To complement these empirical findings, we further provide a theoretical analysis verifying that variations in HE establish a lower bound on the interference with the original pretrained knowledge. This result clarifies that state-of-the-art (SOTA) editing methods implicitly regulate hyperspherical uniformity and the lower bound on the interference, providing a principled explanation for their enhanced robustness.

Motivated by these empirical and theoretical findings, we propose SPHERE (**S**parse **P**rojection for **H**yperspherical **E**nergy-**R**egularized **E**diting), an HE-driven regularization strategy that stabilizes neuron weight distributions, ultimately preserving prior knowledge while enabling reliable sequential updates. The key insight is that, as shown in Figure 1 (b), current methods often introduce perturbations that interfere with the principal hyperspherical directions of the pretrained weight matrices, leading to instability, loss of uniformity, and eventual degradation of model performance. To counteract these side effects, as shown in Figure 1 (c), SPHERE identifies a sparse space complementary to the principal hyperspherical directions of the pretrained weight matrices and projects new knowledge onto it, attenuating perturbation components aligned with those principal directions. By doing so, SPHERE effectively preserves the hyperspherical uniformity and substantially extends the number of effective sequential edits.

To validate the effectiveness of our method, we evaluated SPHERE on **two LLMs**, including LLaMA3 (8B) (AI@Meta, 2024) and Qwen2.5 (7B) (Team, 2024), on **two editing datasets**, including CounterFact (Meng et al., 2022) and ZsRE (Levy et al., 2017). **Four downstream tasks** including reasoning (Cobbe et al., 2021), natural language inference (Dagan et al., 2005), open-domain QA (Kwiatkowski et al., 2019), and closed-domain QA (Clark et al., 2019) are employed to demonstrate the impact of editing on the general abilities of LLMs. Experimental results show that SPHERE sustains editing capacity under large-scale editing settings, outperforming the best baseline (Fang et al., 2025) by **16.41%** on average. Beyond editing capacity, it more effectively preserves the hyperspherical uniformity and the general abilities of edited models than all baselines. Furthermore, as a plug-and-play enhancement, SPHERE improves the editing performance of mainstream methods (Meng et al., 2023; Gu et al., 2024; Ma et al., 2025) by **38.71%** on average, offering a principled path toward reliable and scalable editing. To facilitate others in reproducing our results, we will publish all source code later.

2 PRELIMINARIES

2.1 MODEL EDITING

Sequential model editing aims to update the knowledge stored in LLMs through multiple successive edits. Each edit modifies the model parameter $\boldsymbol{W} \in \mathbb{R}^{d_1 \times d_0}$ by adding a perturbation $\Delta \in \mathbb{R}^{d_1 \times d_0}$ in a locate-then-edit paradigm (Meng et al., 2022), where d_0 and d_1 represent the dimensions of the intermediate and output layers of the feed-forward network (FFN), respectively. Specifically, suppose each edit updates u pieces of knowledge in the form of (subject s, relation r, object o), e.g., (s = *United States*, r = *President of*, o = *Donald Trump*). The perturbed parameter is expected to associate u new *key-value* (k-v) pairs, where k and v encode (s, r) and (o) of the new knowledge, respectively. We can stack these keys and values into matrices as follows:

$$\boldsymbol{K}_1 = [\boldsymbol{k}_1 \,|\, \boldsymbol{k}_2 \,|\, \ldots \,|\, \boldsymbol{k}_u] \in \mathbb{R}^{d_0 \times u}, \quad \boldsymbol{V}_1 = [\boldsymbol{v}_1 \,|\, \boldsymbol{v}_2 \,|\, \ldots \,|\, \boldsymbol{v}_u] \in \mathbb{R}^{d_1 \times u}, \tag{1}$$

where the subscripts of $\boldsymbol{k}$ and $\boldsymbol{v}$ represent the index of the to-be-updated knowledge. Therefore, the editing objective can be expressed as:

$$\Delta \boldsymbol{W} = \arg\min_{\Delta \hat{\boldsymbol{W}}} \left\| (\boldsymbol{W} + \Delta \hat{\boldsymbol{W}}) \boldsymbol{K}_1 - \boldsymbol{V}_1 \right\|^2, \tag{2}$$

where $\| \cdot \|^2$ denotes the sum of the squared elements in the matrix.

Additionally, current methods typically incorporate an error term to preserve the original knowledge. Let $\boldsymbol{K}_0$ and $\boldsymbol{V}_0$ represent the matrices formed by stacking the $\boldsymbol{k}$ and $\boldsymbol{v}$ corresponding to the original pretrained knowledge. Eqn. 2 is regularized by involving the error term as follows:

$$\Delta \boldsymbol{W} = \arg\min_{\Delta \hat{\boldsymbol{W}}} \left(\left\| (\boldsymbol{W} + \Delta \hat{\boldsymbol{W}}) \boldsymbol{K}_1 - \boldsymbol{V}_1 \right\|^2 + \left\| (\boldsymbol{W} + \Delta \hat{\boldsymbol{W}}) \boldsymbol{K}_0 - \boldsymbol{V}_0 \right\|^2 \right). \tag{3}$$

Since $\boldsymbol{K}_0$ and $\boldsymbol{V}_0$ encode the original pretrained knowledge, we have $\boldsymbol{W}\boldsymbol{K}_0 = \boldsymbol{V}_0$ (cf. Eqn. 1). By applying the normal equation, if the closed-form solution of Eqn. 3 exists, it can be written as:

$$\Delta \boldsymbol{W} = (\boldsymbol{V}_1 - \boldsymbol{W}\boldsymbol{K}_1)\boldsymbol{K}_T^\top \left(\boldsymbol{K}_0 \boldsymbol{K}_0^\top + \boldsymbol{K}_1 \boldsymbol{K}_1^\top \right)^{-1}. \tag{4}$$

Since the full scope of an LLM's knowledge is generally inaccessible, $\boldsymbol{K}_0$ is difficult to obtain directly but can be approximated from abundant text input. See Appendix B for more details.

2.2 HYPERSPHERICAL ENERGY

Hyperspherical Energy (HE) serves as a quantitative metric for measuring *hyperspherical uniformity*. Given a group of neurons, HE characterizes their uniformity on a hypersphere by defining a generic potential energy based on their pairwise relationship. Lower energy represents that these neurons are more diverse and uniformly distributed, while higher energy reflects redundancy. Given a weight matrix $\boldsymbol{W} \in \mathbb{R}^{N \times (d+1)}$ represented as a set of N neurons (i.e., kernels), where each row $\boldsymbol{w}_i \in \mathbb{R}^{d+1}$ corresponds to a neuron, its HE is defined as:

$$\boldsymbol{E}_{\boldsymbol{s},\boldsymbol{d}} \left(\hat{\boldsymbol{w}}_i \,|_{i=1}^N \right) = \sum_{i=1}^N \sum_{j=1, j \neq i}^N f_s \left(\|\hat{\boldsymbol{w}}_i - \hat{\boldsymbol{w}}_j\| \right) = \begin{cases} \sum_{i \neq j} \|\hat{\boldsymbol{w}}_i - \hat{\boldsymbol{w}}_j\|^{-s}, & s > 0 \\ \sum_{i \neq j} \log \left(\|\hat{\boldsymbol{w}}_i - \hat{\boldsymbol{w}}_j\|^{-1} \right), & s = 0 \end{cases} \tag{5}$$

where $\| \cdot \|$ denotes Euclidean distance, $f_s(\cdot)$ is a decreasing real-valued function, and $\hat{\boldsymbol{w}}_i = \frac{\boldsymbol{w}_i}{\|\boldsymbol{w}_i\|}$ is the i-th neuron weight projected onto the unit hypersphere $\mathbb{S}^d = \{\boldsymbol{w} \in \mathbb{R}^{d+1} \mid \|\boldsymbol{w}\| = 1\}$. We also denote $\hat{\boldsymbol{W}}_N = \{\hat{\boldsymbol{w}}_1, \cdots, \hat{\boldsymbol{w}}_N \in \mathbb{S}^d\}$, and $E_s = \boldsymbol{E}_{\boldsymbol{s},\boldsymbol{d}}(\hat{\boldsymbol{w}}_i \,|_{i=1}^N)$ for short. There are plenty of choices for $f_s(\cdot)$, but in this paper we use $f_s(z) = z^{-s},\ s > 0$, known as Riesz s-kernels. Since each $\hat{\boldsymbol{w}}_i$ lies on the unit hypersphere, the squared Euclidean distance between two neurons can be equivalently expressed in angular form as $\|\hat{\boldsymbol{w}}_i - \hat{\boldsymbol{w}}_j\|^2 = 2\big(1 - \cos\theta_{ij}\big)$, where θ_{ij} is the angle between $\hat{\boldsymbol{w}}_i$ and $\hat{\boldsymbol{w}}_j$. Substituting this into Eqn. 5, we have:

$$\boldsymbol{E}_{\boldsymbol{s},\boldsymbol{d}} \left(\hat{\boldsymbol{w}}_i \,|_{i=1}^N \right) = \sum_{i=1}^N \sum_{j=1, j \neq i}^N \big(2(1 - \cos\theta_{ij})\big)^{-s/2}. \tag{6}$$

This angular formulation highlights the geometric interpretation of HE: a higher value corresponds to neuron clustering with low angular diversity, while a lower value reflects a more uniform angular distribution across the hypersphere.

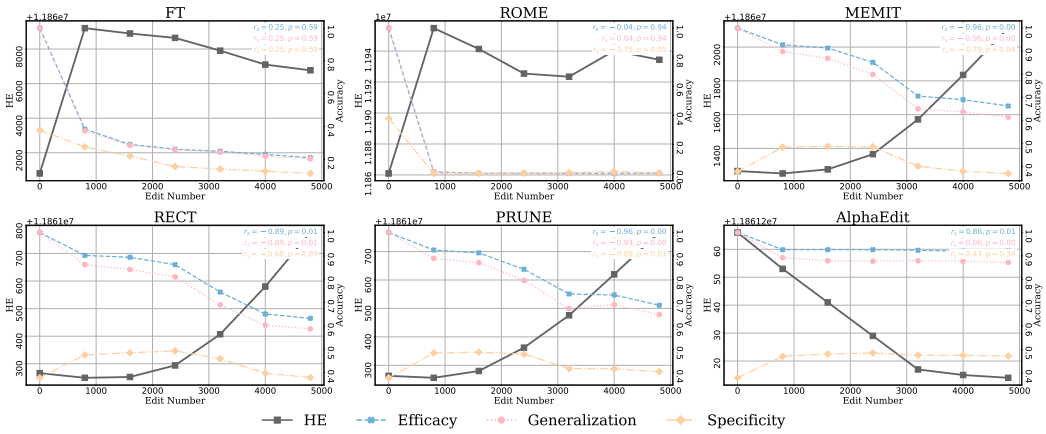


Figure 2: Trends of HE and editing performance throughout sequential editing. The Spearman correlation scores between HE and each editing metric displayed at the end of each curve.

3 Correlation between Hyperspherical Uniformity and Editing

HE and model editing are intrinsically connected through their shared focus on the geometry of high-dimensional parameter spaces. An optimal HE corresponds to more uniformly distributed representations on the unit hypersphere, typically reflecting well-conditioned parameters that enable reliable and stable sequential editing. We first present empirical evidence revealing a strong correlation between HE and editing stability (Section 3.1), followed by a formal theoretical analysis establishing the mathematical link between the two (Section 3.2).

3.1 Empirical Analysis of the HE–Editing Stability Correlation

To understand the failure modes of large-scale sequential editing, we examined how HE evolves throughout the editing process. We performed 5,000 sequential edits on ZsRE dataset (Levy et al., 2017) with a batch size of 100 on LLaMA3-8B (AI@Meta, 2024) using six widely used editing methods, including Fine-Tuning (FT) (Zhu et al., 2020), ROME (Meng et al., 2022), MEMIT (Meng et al., 2023), RECT (Gu et al., 2024), PRUNE (Ma et al., 2025), and AlphaEdit (Fang et al., 2025). After each edit, we computed the HE of the perturbed weights and evaluated the editing performance using well-established metrics, including **Efficacy** (edit success), **Generalization** (paraphrase success), and **Specificity** (neighborhood success). Readers can refer to Appendix D.2 for detailed definition of these metrics. We summarize our main observations as follows:

Observation 1: Collapse in sequential editing is closely tied to sharp fluctuations in HE. Figure 2 reveals a strong correlation between HE dynamics and editing performance. The Spearman correlation scores (Spearman, 1904) between HE and each editing metric, displayed at the end of each curve, consistently indicate a strong statistical dependence before model collapse[1]. Most methods collapse well before 3,000 edits, whereas AlphaEdit demonstrates the strongest long-term editing capacity with the best preservation of hyperspherical uniformity. A closer examination of the metrics shows a consistent pattern in which each drop in performance is consistently accompanied by rapid shifts in HE, underscoring its central role in maintaining sequential editing stability.

Observation 2: Advanced editing methods suppress HE fluctuations effectively. Figure 3 illustrates the correlation between changes in HE (ΔHE) and editing performance (ΔAcc.), where each point denotes the difference between two consecutive batch edits: points near the origin indicate greater stability with minimal variation in both HE and accuracy, while points farther away reflect larger fluctuations and less stable editing. Most advanced methods exhibit tightly clustered distributions near the origin, indicating stable editing dynamics and minimal weight distortion. Furthermore, we fit a linear regression over all points across metrics, which demonstrates a statistically

[1]Since FT and ROME rapidly collapse at the very beginning, we instead emphasize their correlations by examining the curve fluctuations.

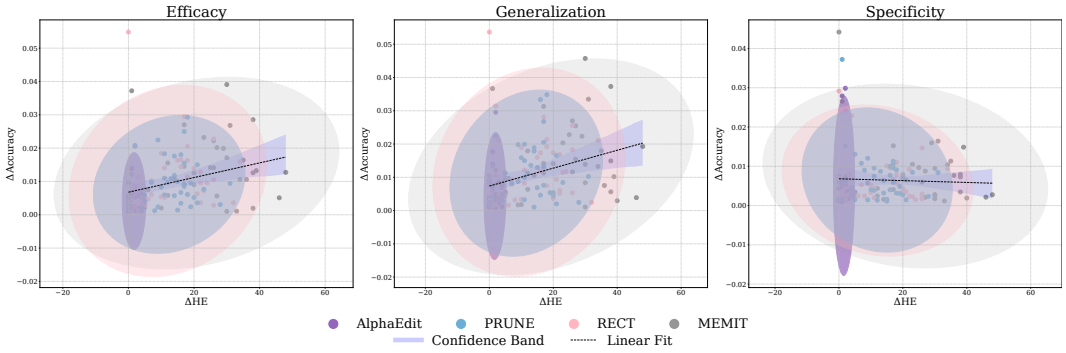


Figure 3: Correlation between changes in HE and editing performance across consecutive edited weights. Each point corresponds to a ΔHE–ΔAcc. pair for one method over five thousand sequential edits. Confidence ellipses and regression lines illustrate overall trends.

significant positive correlation between ΔHE and ΔAcc. in terms of Efficacy and Generalization. This suggests a strong positive correlation between editing stability and HE stability, implying that the effectiveness of SOTA approaches may stem from their ability to suppress HE fluctuations.

3.2 Theoretical Analysis of HE's Impact on Editing Stability

We further turn to a theoretical analysis of how HE impacts editing stability, aiming to provide a principled explanation for the patterns observed in practice. Given the editing objective in Eqn. 2, it inevitably perturbs the original pretrained knowledge in LLMs, which can be expressed as:

$$\Delta \boldsymbol{V} = (\boldsymbol{W} + \Delta \boldsymbol{W})\boldsymbol{K}_1 - \boldsymbol{V}_1 = \Delta \boldsymbol{W}\boldsymbol{K}_1, \tag{7}$$

where $\boldsymbol{W}\boldsymbol{K}_1 = \boldsymbol{V}_1$ (cf. Eqn. 1), as $\boldsymbol{K}_1$ and $\boldsymbol{V}_1$ represent the new editing knowledge. Additionally, from the HE definition in Eqn. 5, the change in HE after editing can be written as:

$$\Delta \boldsymbol{HE} = \sum_{i<j} \left(\|\boldsymbol{w}_i - \boldsymbol{w}_j\|^{-2} - \|\boldsymbol{w}_i + \Delta \boldsymbol{w}_i - \boldsymbol{w}_j - \Delta \boldsymbol{w}_j\|^{-2} \right), \tag{8}$$

where $\Delta \boldsymbol{w}_i$ denotes the perturbation to $\boldsymbol{w}_i$. This term $\Delta \boldsymbol{HE}$ measures how angular separation among weight vectors changes after editing.

Our theoretical analysis, detailed in Appendix C.1, culminates in a key result that formally links the geometric change in weight space $\Delta \boldsymbol{HE}$ to the output perturbation $\Delta \boldsymbol{V}$, derived from Proposition 2.

Theorem 1 (Lower Bound on Output Perturbation)**.** *Under the assumptions of orthonormal inputs and small perturbations, the output perturbation $\Delta \boldsymbol{V}$ is lower-bounded by squared change in HE:*

$$|\Delta \boldsymbol{V}| \;\geq\; \left(\frac{\Delta \boldsymbol{HE}}{K} \right)^2, \quad K = 4 \left(\sum_{k=1}^{p} \left(\sum_{j \neq k} \|\boldsymbol{w}_k - \boldsymbol{w}_j\|^{-3} \right)^2 \right)^{1/2}. \tag{9}$$

where K is a constant dependent on the original weight matrix geometry.

This theorem reveals a key insight: the change in HE ($|\Delta \boldsymbol{HE}|$) inevitably induces a substantial output perturbation ($\Delta \boldsymbol{V}$), meaning that edits that significantly distort the geometric arrangement of neurons are bound to corrupt pretrained knowledge. This result provides a solid theoretical foundation for our empirical findings and underscores HE as a fundamental indicator of editing stability.

4 SPHERE

On account of the above findings, we argue that ideal sequential editing should preserve the hyperspherical uniformity of edited weights. Accordingly, we introduce SPHERE, an HE-driven regularization strategy designed to mitigate HE fluctuations while integrating new knowledge.

SPHERE first estimates the principal hyperspherical directions of pretrained knowledge and then defines their orthogonal complement as the sparse space. Projecting editing perturbations onto this space enables knowledge injection while minimizing interference with original knowledge.

Principal Space Estimation To identify the principal hyperspherical directions in $\boldsymbol{W}$, we seek a unit vector $v \in \mathbb{R}^d$ that maximizes the variance of all neurons in $\boldsymbol{W}$ when projected onto v as:

$$\boldsymbol{v} = \arg\max_{\|\hat{\boldsymbol{v}}\|=1} \left(\tfrac{1}{n}\|\boldsymbol{W}\hat{\boldsymbol{v}}\|^2\right) = \arg\max_{\|\hat{\boldsymbol{v}}\|=1} \left(\tfrac{1}{n}\hat{\boldsymbol{v}}^\top\left(\boldsymbol{W}^\top\boldsymbol{W}\right)\hat{\boldsymbol{v}}\right). \tag{10}$$

According to the Rayleigh quotient theory (Horn & Johnson, 1985; Parlett, 1998), the maximum of $\frac{1}{n}v^\top\boldsymbol{W}^\top\boldsymbol{W}v$ corresponds to the largest eigenvalue λ^* of $\frac{1}{n}\boldsymbol{W}^\top\boldsymbol{W}$, with the associated eigenvector v^* as the principal direction. Extending this to the top-r principal directions enables us to capture a richer low-dimensional space of the weight geometry, so we collect the eigenvectors associated with the r largest eigenvalues to form the principal space matrix, which can be expressed as:

$$\boldsymbol{U} = [\boldsymbol{v}_{d-r+1}, \ldots, \boldsymbol{v}_d] \in \mathbb{R}^{d\times r}, \tag{11}$$

where r satisfies $\sum_{i=d-r+1}^{d}\lambda_i \geq \eta\sum_{i=1}^{d}\lambda_i$, with the cumulative ratio η (see Appendix D.4.1).

Sparse Space Definition This space is defined as the orthogonal complement of $\boldsymbol{U}$ in Eqn. 11 as:

$$\boldsymbol{P}_\perp = I - \alpha\boldsymbol{U}\boldsymbol{U}^\top \in \mathbb{R}^{d\times d}, \tag{12}$$

where α controls the suppression strength of the components along the subspace spanned by $\boldsymbol{U}$ (see Appendix D.4.1). Specifically, $\alpha = 1$ corresponds to a hard orthogonal projection that completely removes the contribution of $\boldsymbol{U}$, while $0 < \alpha < 1$ yields a soft projection that only attenuates it.

Sparse Space Projection Given a perturbation matrix $\Delta\boldsymbol{W}$ produced by any editing method, we project it onto the sparse space using $P_\perp$, and then combine it with the original weight matrix as:

$$\hat{\boldsymbol{W}} = \boldsymbol{W} + \Delta\boldsymbol{W}_{\text{proj}} = \boldsymbol{W} + \Delta\boldsymbol{W}\boldsymbol{P}_\perp. \tag{13}$$

In summary, SPHERE suppresses perturbations aligned with the principal weight directions to preserve hyperspherical uniformity, enabling more stable, longer-lasting performance without compromising general abilities. For theoretical completeness, we also provide a mathematical proof that SPHERE suppresses the $\Delta\boldsymbol{H}\boldsymbol{E}$, ensuring bounded variations in the hidden representations $\Delta\boldsymbol{V}$ and justifying its effectiveness during editing (see Appendix C.2). More details in Appendix D.5

5 Experiments

In this section, we aim to address the following research questions:

- **RQ1:** How does SPHERE perform on sequential editing tasks compared to baseline methods?
- **RQ2:** Can SPHERE effectively preserve the hyperspherical uniformity of edited weights?
- **RQ3:** How does SPHERE-edited LLMs perform on general ability evaluations?
- **RQ4:** Can baseline methods be significantly improved with plug-and-play SPHERE?

5.1 Experimental Setup

Base LLMs and Baseline Methods. Experiments were conducted on LLaMA3 (8B) (AI@Meta, 2024) and Qwen2.5 (7B) (Team, 2024). We compared our approach against a range of representative sequential editing baselines, including Fine-Tuning (FT) (Zhu et al., 2020), MEMIT (Meng et al., 2023), RECT (Gu et al., 2024), PRUNE (Ma et al., 2025), and AlphaEdit (Fang et al., 2025).

Datasets and Evaluation Metrics. Two widely used benchmarks were adopted: CounterFact (Meng et al., 2022) and ZsRE (Levy et al., 2017). Following prior work (Meng et al., 2022), five evaluation metrics were reported: **Efficacy** (edit success), **Generalization** (paraphrase success), **Specificity** (neighborhood success), **Fluency** (generation entropy), and **Consistency** (reference score). For rigorous evaluation, we adopt the **average top-1 accuracy** as the metric for both datasets. Readers can refer to Appendix D for more detailed experimental setup.

5.2 Performance of Sequential Model Editing (RQ1)

Table 1 presents the results under a commonly used sequential editing setup, using 15,000 samples with 100 edits each for LLaMA3 (8B), while Qwen2.5 (7B) is restricted to 5,000 edits as further

Table 1: Comparison of SPHERE with existing methods on sequential editing. *Eff.*, *Gen.*, *Spe.*, *Flu.* and *Consis.* denote Efficacy, Generalization, Specificity, Fluency and Consistency, respectively. The best results are highlighted in bold, while the second-best results are underlined.

Method	Model	ZSRE			Counterfact				
		Eff.↑	Gen.↑	Spe.↑	Eff.↑	Gen.↑	Spe.↑	Flu.↑	Consis.↑
Pre-edited		35.42±0.30	34.17±0.30	38.02±0.27	0.49±0.07	0.44±0.05	18.09±0.24	634.84±0.12	22.06±0.08
FT	LLaMA3-8B	15.27±0.21	14.78±0.21	5.06±0.10	8.40±0.28	2.54±0.13	0.03±0.01	409.80±0.67	19.35±0.13
MEMIT	LLaMA3-8B	0.00±0.00	0.00±0.00	0.06±0.01	0.00±0.00	0.00±0.00	0.00±0.00	318.19±0.24	4.19±0.04
PRUNE	LLaMA3-8B	10.35±0.18	10.08±0.18	9.55±0.15	1.19±0.11	0.34±0.04	0.62±0.03	**618.72±0.08**	**49.24±0.13**
RECT	LLaMA3-8B	0.01±0.00	0.01±0.01	0.04±0.01	0.57±0.08	0.29±0.04	0.10±0.01	438.83±0.18	9.40±0.05
AlphaEdit	LLaMA3-8B	86.64±0.23	81.28±0.28	28.78±0.22	4.37±0.20	1.71±0.10	0.57±0.03	482.36±0.44	4.71±0.04
SPHERE	LLaMA3-8B	**90.01±0.21**	**84.67±0.26**	**45.40±0.29**	**52.89±0.50**	**32.07±0.39**	**5.01±0.10**	551.51±0.53	30.89±0.13
Pre-edited		35.29±0.29	34.10±0.28	38.44±0.27	0.42±0.06	0.46±0.05	15.06±0.20	624.45±0.11	23.02±0.69
FT	Qwen2.5-7B	4.97±0.14	4.58±0.13	4.01±0.11	15.44±0.36	4.63±0.17	1.46±0.05	214.26±0.09	3.15±0.02
MEMIT	Qwen2.5-7B	0.13±0.02	0.12±0.01	0.04±0.01	0.00±0.00	0.00±0.00	0.00±0.00	370.84±0.30	3.59±0.03
PRUNE	Qwen2.5-7B	47.93±0.36	45.50±0.35	**39.20±0.28**	14.30±0.35	11.27±0.26	**6.75±0.12**	**620.74±0.10**	**29.50±0.08**
RECT	Qwen2.5-7B	0.73±0.04	0.75±0.04	0.05±0.07	0.64±0.08	0.19±0.03	0.09±0.01	368.46±0.27	1.35±0.01
AlphaEdit	Qwen2.5-7B	42.01±0.40	39.99±0.39	13.87±0.20	43.92±0.50	24.37±0.36	2.32±0.06	479.83±0.77	4.67±0.07
SPHERE	Qwen2.5-7B	**70.04±0.36**	**65.43±0.37**	27.35±0.26	**60.76±0.49**	**29.24±0.37**	3.83±0.08	612.67±0.22	14.74±0.07

updates induce severe model collapse, where all editing methods underperformed compared to the pre-edit baseline. SPHERE is plug-and-play, and AlphaEdit (Fang et al., 2025) is adopted as the default base method in Table 1 to better illustrate its capability. Additional experiments with other base methods are presented in Section 5.5. Overall, SPHERE consistently outperforms baseline methods across nearly all metrics and base models. In particular, it achieves substantial gains in both **Efficacy** and **Generalization**, with average improvements of **24.19%** and **16.02%**, respectively, over the best baseline. It also maintains notable performance in Fluency and Consistency, indicating its ability to preserve factual accuracy while generating coherent and natural outputs.

5.3 Analysis of Edited Weights (RQ2)

This analysis evaluates whether SPHERE can effectively maintain the hyperspherical uniformity of edited weights. We extracted the edited weights from LLaMA3 after 15,000 sequential edits on CounterFact. As shown in Figure 4, we computed the cosine similarity between each pair of weight neurons and used heatmap to visualize the hyperspherical uniformity before and after editing. In Figure 5, t-SNE (van der Maaten & Hinton, 2008) was used to visualize the normalized neuron distribution in $\boldsymbol{W}$ before and after editing. It can be seen that SPHERE **effectively preserves hyperspherical uniformity after editing, as the cosine similarity among weight neurons remains close to the original distribution**, thereby avoiding directional collapse. Moreover, the pre- and post-edited weights exhibit nearly overlapping distributions, indicating that SPHERE prevents significant shifts in weights and maintains consistency. In contrast, baseline methods such as MEMIT and AlphaEdit induce clear angular concentration in neuron directions, causing neurons to cluster in limited angular regions and significantly reducing hyperspherical directional uniformity. More results on Qwen2.5 are in Appendix E.1.

5.4 Evaluation of General Abilities (RQ3)

To extensively evaluate whether post-edited LLMs can preserve the general abilities, four representative tasks were adopted following Gu et al. (2024), including **Reasoning** on the GSM8K (Cobbe et al., 2021), measured by solve rate. **Natural language inference (NLI)** on the RTE (Dagan et al., 2005), measured by accuracy of two-way classification, **Open-domain QA** on the Natural Questions (Kwiatkowski et al., 2019), measured by exact match (EM) with the reference answer after minor normalization (Chen et al., 2017; Lee et al., 2019). **Closed-domain QA** on BoolQ (Clark et al., 2019), also measured by EM. Figure 6 depicts how performance varies with the number of edited samples across four tasks. We report general performance every 1k edits up to 5k, and every 5k edits thereafter (up to 15k), providing a comprehensive view of the degradation trend. The

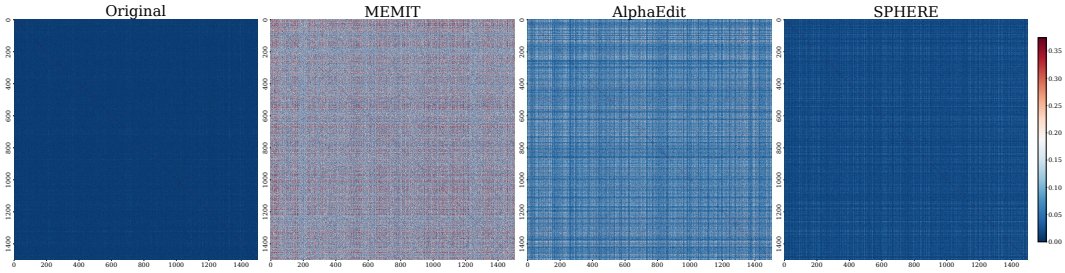


Figure 4: Cosine similarity between neurons in the updated weight matrix after 15,000 edits. Darker colors indicate lower similarity, reflecting better hyperspherical and orthogonal uniformity. SPHERE effectively preserves the weight structure, demonstrating the most stable hyperspherical uniformity.

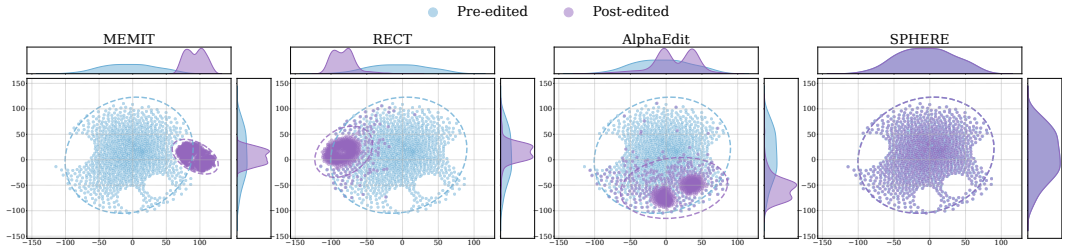


Figure 5: The t-SNE distribution of weight neurons of pre-edited and post-edited LLM after 15,000 edits using dimensionality reduction. The top and right curve graphs display the marginal distributions for two reduced dimensions, where SPHERE consistently exhibits minimal shift.

results show that SPHERE effectively preserves the general abilities of post-edited LLMs even under extensive editing, maintaining the original model performance across all metrics after 15k edits. In contrast, LLMs edited with baseline methods rapidly lose their general abilities with all metrics approaching zero. These findings underscore the critical role of hyperspherical uniformity in safeguarding the broad abilities learned from the underlying corpus.

5.5 Performance Improvements of Baseline Methods (RQ4)

We investigated whether the sparse space projection strategy of SPHERE can serve as a general enhancement to existing methods. A single line of code from SPHERE regarding the projection was inserted into the baselines with minimal modification, and we evaluated their performance before and after integration (as detailed in Appendix D.5). Results of 3,000 sequential edits on LLaMA3 (8B) are reported in Figure 7. **SPHERE is integrated seamlessly with diverse editing methods and significantly boosts their performance.** On average, the optimized baselines achieve relative improvements of **49.05%**, **42.64%**, and **24.44%** in **Efficacy**, **Generalization**, and **Specificity**, respectively, underscoring the strong potential and broad applicability of the proposed sparse space projection as a plug-and-play enhancement for model editing. The baselines enhanced with the projection also demonstrate significantly better robustness in general abilities (see Appendix E.2).

6 Related Work

Model Editing Methods. From the perspective of whether model parameters are modified, existing approaches can be broadly categorized into *parameter-modifying* (Mitchell et al., 2022; Meng et al., 2023; Ma et al., 2025; Fang et al., 2025), which directly adjust a small subset of model parameters, and *parameter-preserving* (Zheng et al., 2023; Yu et al., 2024; Hartvigsen et al., 2023), which integrate auxiliary modules without altering the original weights. In this work, we focus on *parameter-modifying methods* which typically employs meta-learning or locating-then-editing strategies (Zhang et al., 2024b). Representative works of meta-learning include KE (Cao et al., 2021) and MEND (Mitchell et al., 2022), which leverage hypernetworks to generate parameter updates. Locate-then-edit methods, such as ROME (Meng et al., 2022) and MEMIT (Meng et al.,

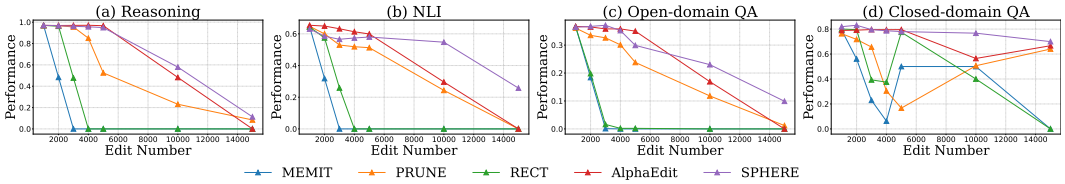
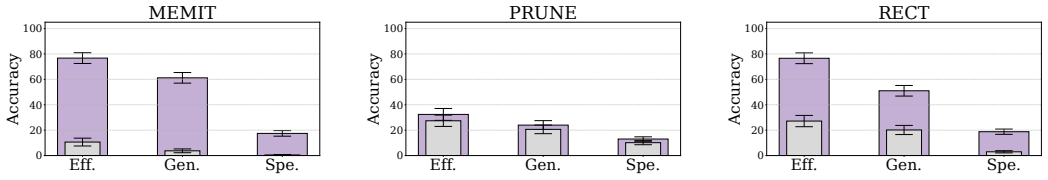
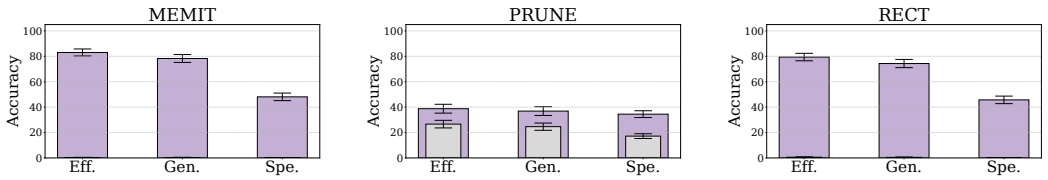


Figure 6: General ability testing of post-edited LLaMA3 (8B) on four tasks.



(a) Performance Improvement on Counterfact Dataset



(b) Performance Improvement on ZsRE Dataset

Figure 7: Performance improvements of baseline editing methods after adding a single line of code from SPHERE (i.e., sparse space projection). Gray bars denote the original baseline performance, while purple bars indicate the performance after enhancement.

2023), prioritize pinpointing the knowledge's storage location before making targeted edits. Recent extensions like RECT (Gu et al., 2024) and PRUNE (Ma et al., 2025) mitigate degradation of general capabilities of LLMs by better constraining edit complexity via sparsity and condition number. Recently, AlphaEdit (Fang et al., 2025) further generalizes this paradigm by projecting the perturbation into the nullspace of the previous knowledge set.

Learning with Hyperspherical Uniformity. Early studies (Xie et al., 2017a; Rodríguez et al., 2017; Xie et al., 2017b; Cogswell et al., 2016) sought to improve the generalization capacity of neural networks by reducing redundancy through diversification, as rigorously analyzed in (Xie et al., 2016). Although these works examined angular diversity, they largely neglected the notion of global equidistribution of embeddings on the hypersphere. In contrast to orthogonality, where perpendicular vectors are defined to be diverse, hyperspherical uniformity promotes embeddings that are maximally separated in angle, thereby encouraging uniform distribution across the hypersphere (Liu et al., 2021; 2018). More recently, Smerkous et al. (2024) enhancing training stability by incorporating centered kernel alignment into hyperspherical energy, enhancing training stability by addressing the lack of permutation invariance inherent in naive similarity metrics.

7 Conclusion

In this work, we demonstrated that hyperspherical uniformity is a critical factor in stabilizing sequential editing for LLMs, supported by both empirical evidence and rigorous theoretical proof. Motivated by this insight, we propose SPHERE, a regularization strategy that preserves hyperspherical uniformity by projecting updates onto a space complementary to the principal directions of pretrained weights. Extensive evaluations on LLaMA3 (8B) and Qwen2.5 (7B) across multiple editing datasets and downstream tasks confirm that SPHERE not only enhances editing performance by 16.41% over the strongest baseline but also more faithfully preserves weight geometry and general abilities of models. Furthermore, when applied as a plug-and-play enhancement, it yields an additional average improvement of 38.71% across existing methods. Collectively, our findings establish SPHERE as both theoretically grounded and empirically effective, providing a principled and scalable solution for reliable large-scale model editing.

Ethics Statement

Our proposed method SPHERE significantly enhances the reliability of large-scale sequential model editing by preserving hyperspherical uniformity, which makes it a valuable way for updating and managing knowledge in real-world applications where long-term stability is essential. At the same time, the ability to directly alter stored knowledge in LLMs carries inherent risks, including the potential introduction of bias or harmful information. To address these concerns, we strongly recommend rigorous validation procedures, transparent reporting, and strict oversight when deploying such techniques. While the core motivation of SPHERE is positive, aiming to facilitate efficient and trustworthy updates of large language models, we emphasize that its use must remain responsible and cautious to ensure ethical outcomes.

We used LLMs to assist with improving grammar, clarity, and wording in parts of this work. The use of LLMs was limited to language refinement, with all ideas, analyses, and conclusions solely developed by the authors. We restate this announcement in Appendix A

Reproducibility Statement

To ensure the reproducibility of our findings, detailed implementation instructions for SPHERE are provided in in Section 4 , Appendix D. Additionally, we plan to release our source code in the future to further support reproducibility. These measures are intended to facilitate the verification and replication of our results by other researchers in the field.

References

AI@Meta. Llama 3 model card, 2024. URL https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md. 1, 3.1, 5.1

Nicola De Cao, Wilker Aziz, and Ivan Titov. Editing factual knowledge in language models. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pp. 6491–6506. Association for Computational Linguistics, 2021. doi: 10.18653/V1/2021.EMNLP-MAIN.522. URL https://doi.org/10.18653/v1/2021.emnlp-main.522. 1, 6

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading wikipedia to answer open-domain questions. In Regina Barzilay and Min-Yen Kan (eds.), *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pp. 1870–1879. Association for Computational Linguistics, 2017. doi: 10.18653/V1/P17-1171. URL https://doi.org/10.18653/v1/P17-1171. 5.4, E.2

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. Boolq: Exploring the surprising difficulty of natural yes/no questions. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pp. 2924–2936. Association for Computational Linguistics, 2019. doi: 10.18653/V1/N19-1300. URL https://doi.org/10.18653/v1/n19-1300. 1, 5.4, E.2

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168, 2021. URL https://arxiv.org/abs/2110.14168. 1, 5.4, E.2

Michael Cogswell, Faruk Ahmed, Ross B. Girshick, Larry Zitnick, and Dhruv Batra. Reducing overfitting in deep networks by decorrelating representations. In Yoshua Bengio and Yann LeCun (eds.), *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016. URL http://arxiv.org/abs/1511.06068. 1, 6

Ido Dagan, Oren Glickman, and Bernardo Magnini. The PASCAL recognising textual entailment challenge. In Joaquin Quiñonero Candela, Ido Dagan, Bernardo Magnini, and Florence d'Alché-Buc (eds.), *Machine Learning Challenges, Evaluating Predictive Uncertainty, Visual Object Classification and Recognizing Textual Entailment, First PASCAL Machine Learning Challenges Workshop, MLCW 2005, Southampton, UK, April 11-13, 2005, Revised Selected Papers*, volume 3944 of *Lecture Notes in Computer Science*, pp. 177–190. Springer, 2005. doi: 10.1007/11736790_9. URL https://doi.org/10.1007/11736790_9. 1, 5.4, E.2

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Litong Wang, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runxin Xu, Ruoyu Zhang, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Shuting Pan, T. Wang, Tao Yun, Tian Pei, Tianyu Sun, W. L. Xiao, and Wangding Zeng. Deepseek-v3 technical report. *CoRR*, abs/2412.19437, 2024. doi: 10.48550/ARXIV.2412.19437. URL https://doi.org/10.48550/arXiv.2412.19437. 1

Junfeng Fang, Houcheng Jiang, Kun Wang, Yunshan Ma, Jie Shi, Xiang Wang, Xiangnan He, and Tat-Seng Chua. Alphaedit: Null-space constrained knowledge editing for language models. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=HvSytvg3Jh. 1, 1, 3.1, 5.1, 5.2, 6, D.5

Jia-Chen Gu, Hao-Xiang Xu, Jun-Yu Ma, Pan Lu, Zhen-Hua Ling, Kai-Wei Chang, and Nanyun Peng. Model editing harms general abilities of large language models: Regularization to the rescue. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pp. 16801–16819. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.EMNLP-MAIN.934. URL https://doi.org/10.18653/v1/2024.emnlp-main.934. 1, 1, 3.1, 5.1, 5.4, 6, E.2

Akshat Gupta, Anurag Rao, and Gopala Anumanchipalli. Model editing at scale leads to gradual and catastrophic forgetting. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pp. 15202–15232. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.FINDINGS-ACL.902. URL https://doi.org/10.18653/v1/2024.findings-acl.902. 1

Tom Hartvigsen, Swami Sankaranarayanan, Hamid Palangi, Yoon Kim, and Marzyeh Ghassemi. Aging with GRACE: lifelong model editing with discrete key-value adaptors. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/95b6e2ff961580e03c0a662a63a71812-Abstract-Conference.html. 6

Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 1985. ISBN 0-521-30586-1. 4

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Trans. Inf. Syst.*, 43(2):42:1–42:55, 2025. doi: 10.1145/3703155. URL https://doi.org/10.1145/3703155. 1

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12):248:1–248:38, 2023. doi: 10.1145/3571730. URL https://doi.org/10.1145/3571730. 1

Houcheng Jiang, Junfeng Fang, Tianyu Zhang, Baolong Bi, An Zhang, Ruipeng Wang, Tao Liang, and Xiang Wang. Neuron-level sequential editing for large language models. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pp. 16678–16702. Association for Computational Linguistics, 2025. URL https://aclanthology.org/2025.acl-long.815/. B

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur P. Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: a benchmark for question answering research. *Trans. Assoc. Comput. Linguistics*, 7:452–466, 2019. doi: 10.1162/TACL_A_00276. URL https://doi.org/10.1162/tacl_a_00276. 1, 5.4, E.2

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. Latent retrieval for weakly supervised open domain question answering. In Anna Korhonen, David R. Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pp. 6086–6096. Association for Computational Linguistics, 2019. doi: 10.18653/V1/P19-1612. URL https://doi.org/10.18653/v1/p19-1612. 5.4, E.2

Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. Zero-shot relation extraction via reading comprehension. In Roger Levy and Lucia Specia (eds.), *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017), Vancouver, Canada, August 3-4, 2017*, pp. 333–342. Association for Computational Linguistics, 2017. doi: 10.18653/V1/K17-1034. URL https://doi.org/10.18653/v1/K17-1034. 1, 3.1, 5.1, D.1

Zherui Li, Houcheng Jiang, Hao Chen, Baolong Bi, Zhenhong Zhou, Fei Sun, Junfeng Fang, and Xiang Wang. Reinforced lifelong editing for language models. *CoRR*, abs/2502.05759, 2025. doi: 10.48550/ARXIV.2502.05759. URL https://doi.org/10.48550/arXiv.2502.05759. B

Weiyang Liu, Rongmei Lin, Zhen Liu, Lixin Liu, Zhiding Yu, Bo Dai, and Le Song. Learning towards minimum hyperspherical energy. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 6225–6236, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/177540c7bcb8db31697b601642eac8d4-Abstract.html. 1, 6

Weiyang Liu, Rongmei Lin, Zhen Liu, Li Xiong, Bernhard Schölkopf, and Adrian Weller. Learning with hyperspherical uniformity. In Arindam Banerjee and Kenji Fukumizu (eds.), *The 24th International Conference on Artificial Intelligence and Statistics, AISTATS 2021, April 13-15, 2021, Virtual Event*, volume 130 of *Proceedings of Machine Learning Research*, pp. 1180–1188. PMLR, 2021. URL http://proceedings.mlr.press/v130/liu21d.html. 1, 6

Jun-Yu Ma, Hong Wang, Hao-Xiang Xu, Zhen-Hua Ling, and Jia-Chen Gu. Perturbation-restrained sequential model editing. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=bfI8cp8qmk. 1, 3.1, 5.1, 6

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/

hash/6f1d43d5a82a37e89b0665b33bf3a182-Abstract-Conference.html. 1, 2.1, 3.1, 5.1, 6, B, D.1, D.2.1, D.2.2

Kevin Meng, Arnab Sen Sharma, Alex J. Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=MkbcAHIYgyS. 1, 1, 3.1, 5.1, 6, B, B, D.2.1, D.2.2, D.4

Meta AI. Introducing llama 4: Advancing multimodal intelligence, 2024. URL https://ai.meta.com/blog/llama-4-multimodal-intelligence/. 1

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D. Manning. Fast model editing at scale. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL https://openreview.net/forum?id=0DcZxeWfOPt. 1, 6, D.2.1

OpenAI. Introducing gpt-5. Online, 2025. URL https://openai.com/index/introducing-gpt-5/. [Large language model announcement]. 1

Beresford N. Parlett. *The Symmetric Eigenvalue Problem*. Classics in Applied Mathematics. SIAM, 1998. ISBN 0-89871-402-8. 4

Zeju Qiu, Weiyang Liu, Haiwen Feng, Yuxuan Xue, Yao Feng, Zhen Liu, Dan Zhang, Adrian Weller, and Bernhard Schölkopf. Controlling text-to-image diffusion by orthogonal finetuning. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/faacb7a4827b4d51e201666b93ab5fa7-Abstract-Conference.html. 1

Pau Rodríguez, Jordi Gonzàlez, Guillem Cucurull, Josep M. Gonfaus, and F. Xavier Roca. Regularizing cnns with locally constrained decorrelations. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL https://openreview.net/forum?id=ByOvsIqeg. 6

David Smerkous, Qinxun Bai, and Fuxin Li. Enhancing diversity in bayesian deep learning via hyperspherical energy minimization of CKA. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/f9e72ee379bb781f3005775c870a3871-Abstract-Conference.html. 6

C Spearman. The proof and measurement of association between two things. *The American Journal of Psychology*, 15(1):72–101, 1904. 3.1

Qwen Team. Qwen2.5: A party of foundation models, September 2024. URL https://qwenlm.github.io/blog/qwen2.5/. 1, 5.1

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. URL http://jmlr.org/papers/v9/vandermaaten08a.html. 5.3

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771, 2019. URL http://arxiv.org/abs/1910.03771. D.4

Pengtao Xie, Jun Zhu, and Eric P. Xing. Diversity-promoting bayesian learning of latent variable models. In Maria-Florina Balcan and Kilian Q. Weinberger (eds.), *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pp. 59–68. JMLR.org, 2016. URL http://proceedings.mlr.press/v48/xiea16.html. 6

Pengtao Xie, Yuntian Deng, Yi Zhou, Abhimanu Kumar, Yaoliang Yu, James Zou, and Eric P. Xing. Learning latent space models with angular constraints. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3799–3810. PMLR, 2017a. URL http://proceedings.mlr.press/v70/xie17a.html. 1, 6

Pengtao Xie, Aarti Singh, and Eric P. Xing. Uncorrelation and evenness: a new diversity-promoting regularizer. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3811–3820. PMLR, 2017b. URL http://proceedings.mlr.press/v70/xie17b.html. 6

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jian Yang, Jiaxi Yang, Jingren Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report. *CoRR*, abs/2505.09388, 2025. doi: 10.48550/ARXIV.2505.09388. URL https://doi.org/10.48550/arXiv.2505.09388. 1

Wanli Yang, Fei Sun, Xinyu Ma, Xun Liu, Dawei Yin, and Xueqi Cheng. The butterfly effect of model editing: Few edits can trigger large language models collapse. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pp. 5419–5437. Association for Computational Linguistics, 2024a. doi: 10.18653/V1/2024.FINDINGS-ACL.322. URL https://doi.org/10.18653/v1/2024.findings-acl.322. B

Wanli Yang, Fei Sun, Jiajun Tan, Xinyu Ma, Du Su, Dawei Yin, and Huawei Shen. The fall of ROME: understanding the collapse of llms in model editing. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024, Miami, Florida, USA, November 12-16, 2024*, pp. 4079–4087. Association for Computational Linguistics, 2024b. doi: 10.18653/V1/2024.FINDINGS-EMNLP.236. URL https://doi.org/10.18653/v1/2024.findings-emnlp.236. B

Lang Yu, Qin Chen, Jie Zhou, and Liang He. MELO: enhancing model editing with neuron-indexed dynamic lora. In Michael J. Wooldridge, Jennifer G. Dy, and Sriraam Natarajan (eds.), *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pp. 19449–19457. AAAI Press, 2024. doi: 10.1609/AAAI.V38I17.29916. URL https://doi.org/10.1609/aaai.v38i17.29916. 6

Ningyu Zhang, Zekun Xi, Yujie Luo, Peng Wang, Bozhong Tian, Yunzhi Yao, Jintian Zhang, Shumin Deng, Mengshu Sun, Lei Liang, Zhiqiang Zhang, Xiaowei Zhu, Jun Zhou, and Huajun Chen. Oneedit: A neural-symbolic collaboratively knowledge editing system. In *Proceedings of Workshops at the 50th International Conference on Very Large Data Bases, VLDB 2024, Guangzhou, China, August 26-30, 2024*. VLDB.org, 2024a. URL https://vldb.org/workshops/2024/proceedings/LLM+KG/LLM+KG-2.pdf. B

Ningyu Zhang, Yunzhi Yao, Bozhong Tian, Peng Wang, Shumin Deng, Mengru Wang, Zekun Xi, Shengyu Mao, Jintian Zhang, Yuansheng Ni, Siyuan Cheng, Ziwen Xu, Xin Xu, Jia-Chen Gu, Yong Jiang, Pengjun Xie, Fei Huang, Lei Liang, Zhiqiang Zhang, Xiaowei Zhu, Jun Zhou, and Huajun Chen. A comprehensive study of knowledge editing for large language models. *CoRR*, abs/2401.01286, 2024b. doi: 10.48550/ARXIV.2401.01286. URL https://doi.org/10.48550/arXiv.2401.01286. 6

Ce Zheng, Lei Li, Qingxiu Dong, Yuxuan Fan, Zhiyong Wu, Jingjing Xu, and Baobao Chang. Can we edit factual knowledge by in-context learning? In Houda Bouamor, Juan Pino, and Kalika

Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pp. 4862–4876. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.EMNLP-MAIN.296. URL https://doi.org/10.18653/v1/2023.emnlp-main.296. 6

Chen Zhu, Ankit Singh Rawat, Manzil Zaheer, Srinadh Bhojanapalli, Daliang Li, Felix X. Yu, and Sanjiv Kumar. Modifying memories in transformer models. *CoRR*, abs/2012.00363, 2020. URL https://arxiv.org/abs/2012.00363. 3.1, 5.1

A USAGE OF LLMS

Throughout the preparation of this manuscript, we used LLMs to assist with improving grammar, clarity, and wording in parts of this work. The use of LLMs was limited to language refinement, with all ideas, analyses, and conclusions solely developed by the authors.

B PRELIMINARIES OF MODEL EDITING

Model editing aims to refine a pre-trained model by applying one or more edits, where each edit replaces a factual association (s, r, o) with new knowledge (s, r, o^*) (Yang et al., 2024b; Li et al., 2025). After editing, the model is expected to recall the updated object o^* when given a natural language prompt $p(s, r)$, such as "The President of the United States is" (Zhang et al., 2024a).

To achieve this, locating-and-editing methods have been proposed for effective model updates (Yang et al., 2024a). These methods typically follow three steps (Jiang et al., 2025):

Step 1: Locating Influential Layers. The first step is to identify the specific FFN layers that encode the target knowledge using causal tracing (Meng et al., 2022). This method involves injecting Gaussian noise into the hidden states and progressively restoring them to their original values. By analyzing the degree to which the original output recovers, the influential layers can be pinpointed as the targets for editing.

Step 2: Acquiring the Expected Output. The second step aims to obtain the desired output of the critical layers identified in Step 1. Following the key–value theory, the key $\boldsymbol{k}$, which encodes (s, r), is processed through the output weights $\boldsymbol{W}_{\text{out}}^l$ to produce the original value $\boldsymbol{v}$ encoding o. Formally,

$$\boldsymbol{k} \triangleq \sigma\big(\boldsymbol{W}_{\text{in}}^l \, \gamma(\boldsymbol{h}^{l-1} + \boldsymbol{\alpha}^l)\big), \qquad \boldsymbol{v} \triangleq \boldsymbol{m}^l = \boldsymbol{W}_{\text{out}}^l k. \tag{14}$$

To perform editing, $\boldsymbol{v}$ is expected to be replaced with a new value $\boldsymbol{v}^*$ encoding o^*. To this end, current methods typically use gradient descent on $\Delta\boldsymbol{W}$, maximizing the probability that the model outputs the word associated with o^* (Meng et al., 2023). The optimization objective is as follows:

$$\boldsymbol{v}^* = \boldsymbol{v} + \arg\min_{\Delta \boldsymbol{W}^l} \Big(-\log \mathbb{P}_{f^l_{\boldsymbol{W}_{\text{out}}}(\boldsymbol{m}^l \mathrel{+}= \Delta \boldsymbol{W}^l)} \big[\, o^* \mid (s, r) \,\big] \Big), \tag{15}$$

where $f^l_{\boldsymbol{W}_{\text{out}}}(\boldsymbol{m}^l \mathrel{+}= \Delta\boldsymbol{W})$ represents the original model with $\boldsymbol{m}^l$ updated to $\boldsymbol{m}^l + \Delta\boldsymbol{W}$.

Step 3: Updating $\boldsymbol{W}_{\text{out}}^l$. This step aims to update the parameters $\boldsymbol{W}_{\text{out}}^l$. It includes a factual set $\{\boldsymbol{K}_1, \boldsymbol{V}_1\}$ containing u new associations, while preserving the set $\{\boldsymbol{K}_0, \boldsymbol{V}_0\}$ containing n original associations. Specifically,

$$\begin{aligned} \boldsymbol{K}_0 &= [\, \boldsymbol{k}_1 \; \boldsymbol{k}_2 \; \cdots \; \boldsymbol{k}_n \,], \quad \boldsymbol{V}_0 = [\, \boldsymbol{v}_1 \; \boldsymbol{v}_2 \; \cdots \; \boldsymbol{v}_n \,], \\ \boldsymbol{K}_1 &= [\, \boldsymbol{k}_{n+1} \; \boldsymbol{k}_{n+2} \; \cdots \; \boldsymbol{k}_{n+u} \,], \quad \boldsymbol{V}_1 = [\, \boldsymbol{v}^*_{n+1} \; \boldsymbol{v}^*_{n+2} \; \cdots \; \boldsymbol{v}^*_{n+u} \,] \end{aligned} \tag{16}$$

where $\boldsymbol{k}$ and $\boldsymbol{v}$ are defined in Eqn. 14 and their subscripts represent the index of the knowledge. Based on these, the objective can be defined as:

$$\tilde{\boldsymbol{W}}_{\text{out}}^l \triangleq \arg\min_{\hat{\boldsymbol{W}}} \left(\sum_{i=1}^{n} \big\| \hat{\boldsymbol{W}} \boldsymbol{k}_i - \boldsymbol{v}_i \big\|^2 + \sum_{i=n+1}^{n+u} \big\| \hat{\boldsymbol{W}} \boldsymbol{k}_i - \boldsymbol{v}_i^* \big\|^2 \right). \tag{17}$$

By applying the normal equation, its closed-form solution can be derived:

$$\tilde{\boldsymbol{W}}_{\text{out}}^l = \big(\boldsymbol{M}_1 - \boldsymbol{W}_{\text{out}}^l \boldsymbol{K}_1\big) \, \boldsymbol{K}_1^\top \big(\boldsymbol{K}_0 \boldsymbol{K}_0^\top + \boldsymbol{K}_1 \boldsymbol{K}_1^\top\big)^{-1} + \boldsymbol{W}_{\text{out}}^l. \tag{18}$$

In practice, model editing methods often update parameters across multiple layers to improve effectiveness. For more details, see (Meng et al., 2023).

C THEORETICAL PROOFS

C.1 PROOF OF CORRELATION BETWEEN HYPERSPHERICAL ENERGY AND EDITING STABILITY

Objective for Preserving Original Knowledge We begin by assuming that the original knowledge base can be expressed as $\{\boldsymbol{k}_i,\ \boldsymbol{v}_i\}\quad i=1,\ldots,N,\quad \boldsymbol{k}_i\in\mathbb{R}^p,\ \boldsymbol{v}_i\in\mathbb{R}^q$, while the new knowledge base is given by $\{\tilde{\boldsymbol{k}}_i,\ \tilde{\boldsymbol{v}}_i\}\quad i=1,\ldots,N,\quad \tilde{\boldsymbol{k}}_i\in\mathbb{R}^p,\ \tilde{\boldsymbol{v}}_i\in\mathbb{R}^q$

And the knowledge mapping is governed by the weight matrix $W\in\mathbb{R}^{p\times q}$ such that

$$\boldsymbol{W}\boldsymbol{k}_i=\boldsymbol{v}_i,\qquad(\boldsymbol{W}+\Delta\boldsymbol{W})\tilde{\boldsymbol{k}}_i=\tilde{\boldsymbol{v}}_i. \tag{19}$$

When analyzing the destroy to the original knowledge set, there is shift brought by the perturbation $\Delta\boldsymbol{W}$ which can be expressed as:

$$(\boldsymbol{W}+\Delta\boldsymbol{W})\boldsymbol{k}_i=\boldsymbol{v}_i+\Delta\boldsymbol{W}\boldsymbol{k}_i \tag{20}$$

where we define $\Delta\boldsymbol{v}_i=\Delta\boldsymbol{W}\boldsymbol{k}_i$ as the destroy to the previous knowledge set. In addition, if $\boldsymbol{k}_i\in\mathrm{null}(\Delta\boldsymbol{W})$, i.e., in the null space of $\Delta\boldsymbol{W}$, then $\Delta\boldsymbol{W}\boldsymbol{k}_i=\Delta\boldsymbol{v}_i=0$.

The corresponding objective is thus to minimize the perturbation magnitude, given by

$$\min\frac{1}{N}\sum_{i=1}^{N}\|\Delta\boldsymbol{v}_i\|\quad\equiv\quad\min\frac{1}{N}\sum_{i=1}^{N}\|\Delta\boldsymbol{W}\boldsymbol{k}_i\|. \tag{21}$$

To make this tractable, assume that each input vector $\boldsymbol{k}_i$ can be approximated in terms of the first $\boldsymbol{K}$ basis vectors $\{\boldsymbol{e_j}\}$ as

$$\boldsymbol{k}_i=\sum_{j=1}^{K}\boldsymbol{\alpha_j}\boldsymbol{e_j}+\varepsilon_i, \tag{22}$$

where ε_i is a small noise term. If we denote

$$\Delta\boldsymbol{W}\cdot\boldsymbol{e_j}=\boldsymbol{f_j},\qquad\varepsilon_i,\boldsymbol{e_j},\boldsymbol{f_j}\in\mathbb{R}^q, \tag{23}$$

then the perturbation objective can be rewritten as:

$$\begin{aligned}
&\min\frac{1}{N}\sum_{i=1}^{N}\left\|\Delta\boldsymbol{W}\cdot\left(\sum_{j=1}^{K}\boldsymbol{\alpha_j}\boldsymbol{e_j}+\varepsilon_i\right)\right\|\\
&\le\frac{1}{N}\sum_{i=1}^{N}\left\|\sum_{j=1}^{K}\boldsymbol{\alpha_j}\boldsymbol{f_j}+\Delta\boldsymbol{W}\varepsilon_i\right\|\\
&\le\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{K}|\boldsymbol{\alpha_j}\boldsymbol{f_j}|+\|\Delta\boldsymbol{W}\||\varepsilon_i|\\
&\le\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{K}|\boldsymbol{\alpha_j}|\,\|\boldsymbol{f_j}\|\ +\ \varepsilon_{\max}\|\boldsymbol{\Delta}\|.
\end{aligned} \tag{24}$$

where $\varepsilon_{\max}=\max_i\|\varepsilon_i\|$. This shows that minimizing $\|f_j\|$ directly reduces the upper bound of the perturbation, and therefore enhances the stability of the editing process.

Definitions Let the model's weights be represented by a matrix $\boldsymbol{W}\in\mathbb{R}^{p\times q}$, whose rows are the neuron vectors $\boldsymbol{w}_1,\ldots,\boldsymbol{w}_p\in\mathbb{R}^q$. An edit or update introduces a perturbation to this matrix, denoted by $\Delta\boldsymbol{W}\in\mathbb{R}^{p\times q}$, with corresponding row-wise perturbations $\Delta\boldsymbol{w}_1,\ldots,\Delta\boldsymbol{w}_p$.

We define two key scalar quantities to measure the effects of this perturbation:

- **output perturbation ($\Delta \boldsymbol{V}$).** This quantity measures the total squared change in the model's output, aggregated over a set of N input vectors $\{\boldsymbol{k}_i\}_{i=1}^N$.

$$\Delta \boldsymbol{V} \triangleq \sum_{i=1}^{N} \|\Delta \boldsymbol{W} \boldsymbol{k}_i\|_2^2 = \sum_{i=1}^{N} \left\| \begin{bmatrix} \Delta \boldsymbol{w}_1 \cdot \boldsymbol{k}_i \\ \vdots \\ \Delta \boldsymbol{w}_p \cdot \boldsymbol{k}_i \end{bmatrix} \right\|_2^2 = \sum_{i=1}^{N} \sum_{j=1}^{p} (\Delta \boldsymbol{w}_j \cdot \boldsymbol{k}_i)^2 \tag{25}$$

- **Change in Hyperdimensional Energy ($\Delta \boldsymbol{HE}$).** This quantity measures the change in the geometric arrangement of the neuron vectors due to the perturbation.

$$\Delta \boldsymbol{HE} \triangleq \sum_{i \neq j} \left(\frac{1}{\|\boldsymbol{w}_i - \boldsymbol{w}_j\|_2^2} - \frac{1}{\|(\boldsymbol{w}_i + \Delta \boldsymbol{w}_i) - (\boldsymbol{w}_j + \Delta \boldsymbol{w}_j)\|_2^2} \right) \tag{26}$$

Assumptions Our analysis relies on the following assumptions:

Assumption 1 (Orthonormal Inputs)**.** *The set of input vectors $\{\boldsymbol{k}_i\}_{i=1}^q$ is the standard orthonormal basis of $\mathbb{R}^q$.*

Under this assumption, the output perturbation simplifies to the squared Frobenius norm of the perturbation matrix:

$$\Delta \boldsymbol{V} = \sum_{i=1}^{q} \sum_{j=1}^{p} (\Delta \boldsymbol{w}_j \cdot \boldsymbol{k}_i)^2 = \sum_{j=1}^{p} \sum_{i=1}^{q} (\Delta \boldsymbol{w}_{j,i})^2 = \sum_{j=1}^{p} \|\Delta \boldsymbol{w}_j\|_2^2 = \|\Delta \boldsymbol{W}\|_F^2$$

Assumption 2 (Small Perturbations)**.** *The perturbation vectors $\Delta \boldsymbol{w}_i$ are sufficiently small in norm, which justifies the use of a first-order Taylor expansion to approximate the change in HE.*

We can now state the relationship between the change in HE and the output perturbation energy.

Theorem 2 (Upper Bound on HE Change)**.** *Under Assumptions 1 and 2, the absolute change in Hyperdimensional Energy, $|\Delta \boldsymbol{HE}|$, is upper-bounded by the square root of the output perturbation, $\sqrt{\Delta \boldsymbol{V}}$, as follows:*

$$|\Delta \boldsymbol{HE}| \leq K \sqrt{\Delta \boldsymbol{V}} \tag{27}$$

where K is a constant determined by the geometry of the original weight matrix $\boldsymbol{W}$:

$$K = 4 \sqrt{\sum_{k=1}^{p} \left(\sum_{j \neq k} \|\boldsymbol{w}_k - \boldsymbol{w}_j\|^{-3} \right)^2}$$

Proof. Let $\boldsymbol{p}_{ij} = \boldsymbol{w}_i - \boldsymbol{w}_j$ and $\Delta \boldsymbol{p}_{ij} = \Delta \boldsymbol{w}_i - \Delta \boldsymbol{w}_j$. The change in HE is $\Delta \boldsymbol{HE} = \sum_{i \neq j} (\|\boldsymbol{p}_{ij}\|^{-2} - \|\boldsymbol{p}_{ij} + \Delta \boldsymbol{p}_{ij}\|^{-2})$. Using a first-order Taylor expansion for $f(\boldsymbol{x}) = \|\boldsymbol{x}\|^{-2}$ around $\boldsymbol{p}_{ij}$, we have:

$$\|\boldsymbol{p}_{ij} + \Delta \boldsymbol{p}_{ij}\|^{-2} \approx \|\boldsymbol{p}_{ij}\|^{-2} - 2\|\boldsymbol{p}_{ij}\|^{-4} (\boldsymbol{p}_{ij} \cdot \Delta \boldsymbol{p}_{ij})$$

Substituting this into the expression for $\Delta \boldsymbol{HE}$:

$$\begin{aligned} \Delta \boldsymbol{HE} &\approx \sum_{i \neq j} \left(\|\boldsymbol{p}_{ij}\|^{-2} - \left(\|\boldsymbol{p}_{ij}\|^{-2} - 2\|\boldsymbol{p}_{ij}\|^{-4} (\boldsymbol{p}_{ij} \cdot \Delta \boldsymbol{p}_{ij}) \right) \right) \\ &= \sum_{i \neq j} 2\|\boldsymbol{p}_{ij}\|^{-4} (\boldsymbol{p}_{ij} \cdot \Delta \boldsymbol{p}_{ij}) \end{aligned}$$

We bound the absolute value of this approximation:

$$\begin{aligned} |\Delta \boldsymbol{HE}| &\approx \left| \sum_{i \neq j} 2\|\boldsymbol{w}_i - \boldsymbol{w}_j\|^{-4} ((\boldsymbol{w}_i - \boldsymbol{w}_j) \cdot (\Delta \boldsymbol{w}_i - \Delta \boldsymbol{w}_j)) \right| \\ &\leq \sum_{i \neq j} 2\|\boldsymbol{w}_i - \boldsymbol{w}_j\|^{-3} \|\Delta \boldsymbol{w}_i - \Delta \boldsymbol{w}_j\| && \text{(by Cauchy-Schwarz)} \\ &\leq \sum_{i \neq j} 2\|\boldsymbol{w}_i - \boldsymbol{w}_j\|^{-3} (\|\Delta \boldsymbol{w}_i\| + \|\Delta \boldsymbol{w}_j\|) && \text{(by Triangle Inequality)} \end{aligned}$$

By re-indexing the sum to group terms by $\|\Delta \boldsymbol{w}_k\|$:

$$|\Delta \boldsymbol{HE}| \leq \sum_{k=1}^{p} \left(\sum_{j \neq k} 2\|\boldsymbol{w}_k - \boldsymbol{w}_j\|^{-3} + \sum_{i \neq k} 2\|\boldsymbol{w}_i - \boldsymbol{w}_k\|^{-3} \right) \|\Delta \boldsymbol{w}_k\|$$
$$= \sum_{k=1}^{p} \left(4 \sum_{j \neq k} \|\boldsymbol{w}_k - \boldsymbol{w}_j\|^{-3} \right) \|\Delta \boldsymbol{w}_k\|$$

Applying the Cauchy-Schwarz inequality to this final sum (viewed as a dot product in $\mathbb{R}^p$):

$$|\Delta \boldsymbol{HE}| \leq \sqrt{\sum_{k=1}^{p} \left(4 \sum_{j \neq k} \|\boldsymbol{w}_k - \boldsymbol{w}_j\|^{-3} \right)^2} \cdot \sqrt{\sum_{k=1}^{p} \|\Delta \boldsymbol{w}_k\|^2}$$
$$= K \cdot \sqrt{\sum_{k=1}^{p} \|\Delta \boldsymbol{w}_k\|^2}$$

From Assumption 1, we know $\sum_{k=1}^{p} \|\Delta \boldsymbol{w}_k\|^2 = \Delta \boldsymbol{V}$. Therefore:

$$|\Delta \boldsymbol{HE}| \leq K\sqrt{\Delta \boldsymbol{V}}$$

□

C.2 Proof of Correlation Between Sparse Space Projection and Hyperspherical Energy

Lemma 1. *For any vector $\boldsymbol{x} \in \mathbb{R}^d$ and a small perturbation $\Delta \boldsymbol{x} \in \mathbb{R}^d$, the first-order Taylor expansion of the function $g(\boldsymbol{x}) = \|\boldsymbol{x}\|_2^{-s}$ is:*

$$g(\boldsymbol{x} + \Delta \boldsymbol{x}) \approx g(\boldsymbol{x}) + \nabla g(\boldsymbol{x})^T \Delta \boldsymbol{x} = \|\boldsymbol{x}\|_2^{-s} - s\|\boldsymbol{x}\|_2^{-s-2} x^T \Delta \boldsymbol{x}. \tag{28}$$

We have

$$g(\boldsymbol{x}) = \left(\sum_k \boldsymbol{x_k^2} \right)^{-s/2}. \tag{29}$$

The partial derivative with respect to x_l is:

$$\frac{\partial g}{\partial \boldsymbol{x}_l} = -\frac{s}{2} \left(\sum_k \boldsymbol{x}_k^2 \right)^{-s/2-1} \cdot (2\boldsymbol{x}_l) = -s\|\boldsymbol{x}\|_2^{-s-2} \boldsymbol{x}_l. \tag{30}$$

Thus, the gradient vector is

$$\nabla g(\boldsymbol{x}) = -s\|\boldsymbol{x}\|_2^{-s-2} \boldsymbol{x}. \tag{31}$$

Substituting into the first-order Taylor expansion

$$g(\boldsymbol{x} + \Delta \boldsymbol{x}) \approx g(\boldsymbol{x}) + \nabla g(\boldsymbol{x})^T \Delta \boldsymbol{x} \tag{32}$$

completes the proof.

Theorem 3. *The magnitude of $|\Delta \boldsymbol{HE}|$ is bounded above by a constant-weighted sum of all neuron perturbation norms:*

$$|\Delta \boldsymbol{HE}| \leq \sum_{k=1}^{p} C_k \|\Delta \boldsymbol{w}_k\|_2, \tag{33}$$

where

$$C_k = s \sum_{j \neq k} \|\boldsymbol{w}_k - \boldsymbol{w}_j\|_2^{-s-1} \tag{34}$$

is a constant that depends only on the original weight matrix $\boldsymbol{W}$.

Consider each term in $\Delta \boldsymbol{HE}$. Let

$$\boldsymbol{p}_{ij} = \boldsymbol{w}_i - \boldsymbol{w}_j, \quad \Delta \boldsymbol{p}_{ij} = (\boldsymbol{w}_i' - \boldsymbol{w}_j') - \boldsymbol{p}_{ij} = \Delta \boldsymbol{w}_i - \Delta \boldsymbol{w}_j. \tag{35}$$

Then

$$\Delta \boldsymbol{HE} = \sum_{i<j} \Big(\|\boldsymbol{p}_{ij}\|_2^{-s} - \|\boldsymbol{p}_{ij} + \Delta \boldsymbol{p}_{ij}\|_2^{-s} \Big). \tag{36}$$

By Lemma 1:

$$\|\boldsymbol{p}_{ij} + \Delta \boldsymbol{p}_{ij}\|_2^{-s} \approx \|\boldsymbol{p}_{ij}\|_2^{-s} - s\|\boldsymbol{p}_{ij}\|_2^{-s-2} \boldsymbol{p}_{ij}^T \Delta \boldsymbol{p}_{ij}. \tag{37}$$

Substituting into the expression for ΔHE:

$$\Delta \boldsymbol{HE} \approx \sum_{i<j} s\|\boldsymbol{p}_{ij}\|_2^{-s-2} \boldsymbol{p}_{ij}^T \Delta \boldsymbol{p}_{ij}. \tag{38}$$

To obtain a rigorous bound, apply the mean value theorem. For $g(\boldsymbol{x}) = \|\boldsymbol{x}\|_2^{-s}$, there exists ξ_{ij} between $\boldsymbol{p}_{ij}$ and $\boldsymbol{p}_{ij} + \Delta \boldsymbol{p}_{ij}$ such that

$$g(\boldsymbol{p}_{ij} + \Delta \boldsymbol{p}_{ij}) - g(\boldsymbol{p}_{ij}) = \nabla g(\xi_{ij})^T \Delta \boldsymbol{p}_{ij}. \tag{39}$$

Taking absolute values and applying the Cauchy–Schwarz inequality:

$$|g(\boldsymbol{p}_{ij} + \Delta \boldsymbol{p}_{ij}) - g(\boldsymbol{p}_{ij})| \leq \|\nabla g(\xi_{ij})\|_2 \cdot \|\Delta \boldsymbol{p}_{ij}\|_2. \tag{40}$$

Since

$$\nabla g(\boldsymbol{x}) = -s\|\boldsymbol{x}\|_2^{-s-2} \boldsymbol{x}, \tag{41}$$

its norm is

$$\|\nabla g(\boldsymbol{x})\|_2 = s\|\boldsymbol{x}\|_2^{-s-1}. \tag{42}$$

Assuming small perturbations, $\xi_{ij} \approx \boldsymbol{p}_{ij}$, giving

$$|g(\boldsymbol{p}_{ij} + \Delta \boldsymbol{p}_{ij}) - g(\boldsymbol{p}_{ij})| \approx s\|\boldsymbol{p}_{ij}\|_2^{-s-1} \|\Delta \boldsymbol{p}_{ij}\|_2. \tag{43}$$

Thus,

$$|\Delta HE| \leq \sum_{i<j} s\|\boldsymbol{w}_i - \boldsymbol{w}_j\|_2^{-s-1} \|\Delta \boldsymbol{p}_{ij}\|_2. \tag{44}$$

Applying the triangle inequality:

$$\|\Delta \boldsymbol{p}_{ij}\|_2 = \|\Delta \boldsymbol{w}_i - \Delta \boldsymbol{w}_j\|_2 \leq \|\Delta \boldsymbol{w}_i\|_2 + \|\Delta \boldsymbol{w}_j\|_2. \tag{45}$$

Therefore,

$$|\Delta \boldsymbol{HE}| \leq \sum_{i<j} s\|\boldsymbol{w}_i - \boldsymbol{w}_j\|_2^{-s-1} (\|\Delta \boldsymbol{w}_i\|_2 + \|\Delta \boldsymbol{w}_j\|_2). \tag{46}$$

Rearranging terms with respect to each $\|\Delta \boldsymbol{w}_k\|_2$, we obtain:

$$|\Delta \boldsymbol{HE}| \leq \sum_{k=1}^{p} \|\Delta \boldsymbol{w}_k\|_2 \Big(s \sum_{j \neq k} \|\boldsymbol{w}_k - \boldsymbol{w}_j\|_2^{-s-1} \Big). \tag{47}$$

Defining

$$C_k = s \sum_{j \neq k} \|\boldsymbol{w}_k - \boldsymbol{w}_j\|_2^{-s-1}, \tag{48}$$

we conclude

$$|\Delta \boldsymbol{HE}| \leq \sum_{k=1}^{p} C_k \|\Delta \boldsymbol{w}_k\|_2. \tag{49}$$

Conclusion of Theorem 1. The magnitude of $|\Delta \boldsymbol{HE}|$ is constrained by the weighted sum of neuron perturbation norms. To reduce $|\Delta \boldsymbol{HE}|$, an effective approach is to minimize each $\|\Delta \boldsymbol{w}_k\|_2$.

Theorem 4. *The SPHERE projection operation reduces (or preserves) the ℓ_2-norm of perturbation vectors:*

$$\|\Delta \boldsymbol{w}_{i,\textit{SPHERE}}\|_2 \leq \|\Delta \boldsymbol{w}_i\|_2. \tag{50}$$

Compute the squared ℓ_2-norm:

$$\|\Delta \boldsymbol{w}_{i,\text{SPHERE}}\|_2^2 = \|\Delta \boldsymbol{w}_i \boldsymbol{P}_\perp\|_2^2 = (\Delta \boldsymbol{w}_i \boldsymbol{P}_\perp)(\Delta \boldsymbol{w}_i \boldsymbol{P}_\perp)^T. \tag{51}$$

Using $(AB)^T = B^T A^T$:

$$\|\Delta \boldsymbol{w}_{i,\text{SPHERE}}\|_2^2 = \Delta \boldsymbol{w}_i \boldsymbol{P}_\perp \boldsymbol{P}_\perp^T \Delta \boldsymbol{w}_i^T. \tag{52}$$

The projection matrix $\boldsymbol{P}_\perp$ satisfies two key properties:

- **Symmetry:** $\boldsymbol{P}_\perp^T = \boldsymbol{P}_\perp$.
- **Idempotence:** $\boldsymbol{P}_\perp^2 = \boldsymbol{P}_\perp$.

Thus,

$$\|\Delta \boldsymbol{w}_{i,\text{SPHERE}}\|_2^2 = \Delta \boldsymbol{w}_i \boldsymbol{P}_\perp^2 \Delta \boldsymbol{w}_i^T = \Delta \boldsymbol{w}_i \boldsymbol{P}_\perp \Delta \boldsymbol{w}_i^T. \tag{53}$$

Substituting $\boldsymbol{P}_\perp = I - \boldsymbol{U}\boldsymbol{U}^T$:

$$\|\Delta \boldsymbol{w}_{i,\text{SPHERE}}\|_2^2 = \Delta \boldsymbol{w}_i (I - \boldsymbol{U}\boldsymbol{U}^T) \Delta \boldsymbol{w}_i^T = \|\Delta \boldsymbol{w}_i\|_2^2 - \|\Delta \boldsymbol{w}_i \boldsymbol{U}\|_2^2. \tag{54}$$

Since

$$\|\Delta \boldsymbol{w}_i \boldsymbol{U}\|_2^2 \geq 0, \tag{55}$$

we conclude

$$\|\Delta \boldsymbol{w}_{i,\text{SPHERE}}\|_2^2 \leq \|\Delta \boldsymbol{w}_i\|_2^2. \tag{56}$$

Taking square roots:

$$\|\Delta \boldsymbol{w}_{i,\text{SPHERE}}\|_2 \leq \|\Delta \boldsymbol{w}_i\|_2. \tag{57}$$

Equality holds iff

$$\Delta \boldsymbol{w}_i \boldsymbol{U} = 0, \tag{58}$$

i.e., $\Delta \boldsymbol{w}_i$ is orthogonal to all basis vectors of the principal subspace $\boldsymbol{U}$. In this case, $\Delta \boldsymbol{w}_i$ already lies in the sparse subspace.

D Experimental Setup

D.1 Datasets

Here, we provide a detailed introduction to the datasets used in this paper:

- **Counterfact** (Meng et al., 2022) is a more challenging dataset that contrasts counterfactual with factual statements, initially scoring lower for Counterfact. It constructs out-of-scope data by replacing the subject entity with approximate entities sharing the same predicate. The Counterfact dataset has similar metrics to ZsRE for evaluating efficacy, generalization, and specificity. Additionally, Counterfact includes multiple generation prompts with the same meaning as the original prompt to test the quality of generated text, specifically focusing on fluency and consistency.
- **ZsRE** (Levy et al., 2017) is a question answering (QA) dataset that uses questions generated through back-translation as equivalent neighbors. Following previous work, natural questions are used as out-of-scope data to evaluate locality. Each sample in ZsRE includes a subject string and answers as the editing targets to assess editing success, along with the rephrased question for generalization evaluation and the locality question for evaluating specificity.

D.2 Evaluation Metrics

Now we introduce the evaluation metrics for the ZsRE and Counterfact datasets, respectively.

D.2.1 METRICS FOR ZSRE

Following the previous work (Mitchell et al., 2022; Meng et al., 2022; 2023), this section defines each ZsRE metric given a LLM f_θ, a knowledge fact prompt (s_i, r_i), an edited target output o_i, and the model's original output o_i^c:

- **Efficacy:** Efficacy is calculated as the average top-1 accuracy on the edit samples:

$$\mathbb{E}_i\Big\{o_i = \arg\max_o \mathbb{P}_{f_\theta}(o \mid (s_i, r_i))\Big\}. \tag{59}$$

- **Generalization:** Generalization measures the model's performance on equivalent prompts of (s_i, r_i), such as rephrased statements $N((s_i, r_i))$. This is evaluated by the average top-1 accuracy on these $N((s_i, r_i))$:

$$\mathbb{E}_i\Big\{o_i = \arg\max_o \mathbb{P}_{f_\theta}(o \mid N((s_i, r_i)))\Big\}. \tag{60}$$

- **Specificity:** Specificity ensures that the editing does not affect samples unrelated to the edit cases $O(s_i, r_i)$. This is evaluated by the top-1 accuracy of predictions that remain unchanged:

$$\mathbb{E}_i\Big\{o_i^c = \arg\max_o \mathbb{P}_{f_\theta}(o \mid O((s_i, r_i)))\Big\}. \tag{61}$$

D.2.2 METRICS FOR COUNTERFACT

Following previous work (Meng et al., 2022; 2023), this section defines the Counterfact metrics given a language model f_θ, a knowledge fact prompt (s_i, r_i), an edited target output o_i, and the model's original output o_i^c. However, for rigorous evaluation, we adopt the **average top-1 accuracy** as the metric for this dataset, which is used to assess Efficacy, Generalization, and Specificity.

- **Efficacy (efficacy success):** Efficacy is calculated as the average top-1 accuracy on the edit samples:

$$\mathbb{E}_i\Big\{o_i = \arg\max_o \mathbb{P}_{f_\theta}(o \mid (s_i, r_i))\Big\}. \tag{62}$$

- **Generalization (paraphrase success):** Generalization measures the model's performance on equivalent prompts of (s_i, r_i), such as rephrased statements $N((s_i, r_i))$. This is evaluated by the average top-1 accuracy on these $N((s_i, r_i))$:

$$\mathbb{E}_i\Big\{o_i = \arg\max_o \mathbb{P}_{f_\theta}(o \mid N((s_i, r_i)))\Big\}. \tag{63}$$

- **Specificity (neighborhood success):** Specificity ensures that the editing does not affect samples unrelated to the edit cases $O(s_i, r_i)$. This is evaluated by the top-1 accuracy of predictions that remain unchanged:

$$\mathbb{E}_i\Big\{o_i^c = \arg\max_o \mathbb{P}_{f_\theta}(o \mid O((s_i, r_i)))\Big\}. \tag{64}$$

- **Fluency (generation entropy):** Measure for excessive repetition in model outputs. It uses the entropy of n-gram distributions:

$$-\frac{2}{3}\sum_k g_2(k)\log_2 g_2(k) + \frac{4}{3}\sum_k g_3(k)\log_2 g_3(k), \tag{65}$$

  where $g_n(\cdot)$ is the n-gram frequency distribution.

- **Consistency (reference score):** The consistency of the model's outputs is evaluated by giving the model f_θ a subject s and computing the cosine similarity between the TF-IDF vectors of the model-generated text and a reference Wikipedia text about o.

D.3 BASELINES

We introduce the five baseline models employed in this study. **For the hyperparameter settings of the baseline methods, except those mentioned in Appendix D.4, we follow the original code provided in the respective papers for reproduction**.

- **MEMIT** is a scalable multi-layer editing algorithm designed to insert new factual memories into transformer-based language models. Extending ROME, MEMIT targets transformer module weights that mediate factual recall, allowing efficient updates of thousands of associations with improved scalability.
- **PRUNE** preserves the general abilities of LLMs during sequential editing by constraining numerical sensitivity. It addresses performance degradation from repeated edits by applying condition number restraints to the edited matrix, thereby limiting harmful perturbations to stored knowledge and ensuring edits can be made without compromising overall model capability.
- **RECT** mitigates unintended side effects of model editing on general reasoning and question answering. It regularizes weight updates during editing to prevent excessive alterations that cause overfitting, thereby maintaining strong editing performance while preserving the model's broader generalization abilities.
- **AlphaEdit** introduces a sequential editing framework that leverages null-space projection to constrain parameter updates. By projecting edits into the null space of unrelated knowledge, AlphaEdit reduces interference with pre-existing capabilities and improves stability under sequential edits. This design enables efficient large-scale editing with enhanced robustness and generalization compared to prior approaches.

D.4 IMPLEMENTATION DETAILS

Our implementation of SPHERE with Llama3 (8B) and Qwen2.5 (7B) follows the configurations outlined in MEMIT (Meng et al., 2023). Specifically, we edit critical layers [4, 5, 6, 7, 8], with the hyperparameters η set to 0.5 and α set to 0.5 (see Appendix D.4.1). During hidden representation updates of the critical layer, we perform 25 optimization steps. The learning rate were set to 0.1 for Llama3 (8B) and 0.5 for Qwen2.5 (7B), respectively. All experiments are conducted on eight A800 (80GB) GPUs. The LLMs are loaded using HuggingFace Transformers (Wolf et al., 2019).

D.4.1 CUMULATIVE RATIO η AND SUPPRESSION STRENGTH α

We next provide details of two important hyperparameters in our sparse space projection: the cumulative ratio η and the suppression strength α, together with the values used in our experiments.

Cumulative Ratio η. We define η as the cumulative ratio used to select the top r eigenvectors in Eqn. 66, corresponding to the r principal directions on the unit hypersphere. Specifically, η controls the selection of eigenvectors based on their eigenvalues λ, such that

$$\sum_{i=d-r+1}^{d} \lambda_i \;\geq\; \eta \cdot \sum_{i=1}^{d} \lambda_i.$$

In practice, we set $\eta = 0.5$ for all experiments, meaning that only the top 50% of the principal directions of the edited weights are suppressed.

$$\boldsymbol{U} = [v_{d-r+1}, \ldots, v_d] \in \mathbb{R}^{d \times r}. \tag{66}$$

Suppression Strength α. We define α as the suppression strength in the projection, which controls the extent to which perturbation components along the principal directions $\boldsymbol{U}$ are removed, as shown in Eqn. 67. In practice, we set $\alpha = 0.5$ for projections on AlphaEdit, while using $\alpha = 0.8$ for all other methods, following the empirical findings reported in Section 3.1 (Observation 2).

$$\boldsymbol{P}_{\perp} = I - \alpha \boldsymbol{U}\boldsymbol{U}^{\top} \in \mathbb{R}^{d \times d}. \tag{67}$$

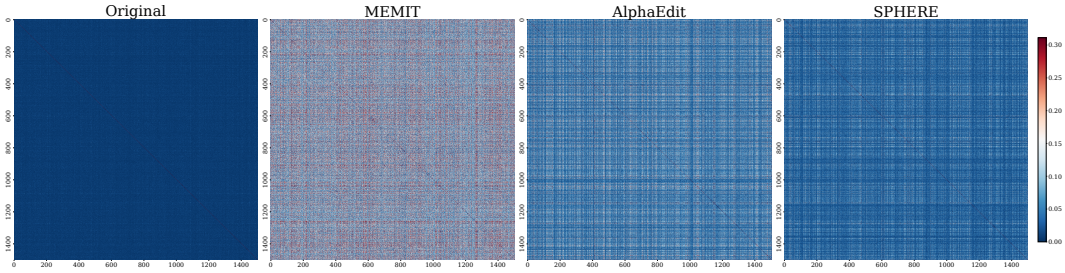


Figure 8: Cosine similarity between neurons in updated weight matrix after 5,000 edits on Qwen2.5. Darker colors indicate lower similarity, reflecting better hyperspherical and orthogonal uniformity. SPHERE effectively preserve the weight structure, demonstrating the most stable hyperspherical uniformity.

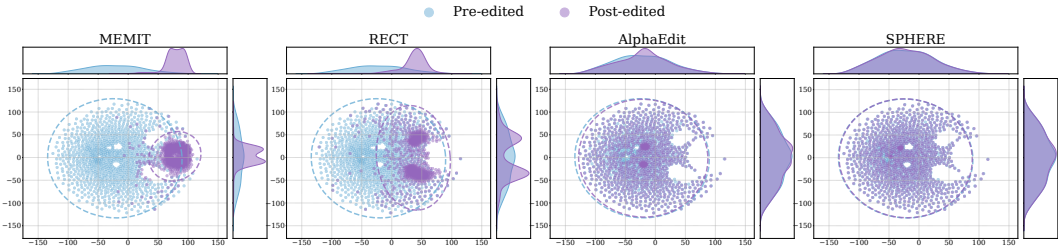


Figure 9: The distribution of weight neurons of pre-edited and post-edited Qwen2.5 after 5,000 edits using dimensionality reduction across mainstream sequential editing methods. The top and right curve graphs display the marginal distributions for two reduced dimensions, where SPHERE consistently exhibits minimal shift.

D.5 Adding Projection in Baseline Methods

We then describe the details of incorporating our projection into baseline editing methods. For illustration, we take MEMIT as an example, though the same procedure is applied to all other methods ($i.e.$ FT, PRUNE, RECT, and AlphaEdit).

As introduced in Section 2.1, the editing objective can be written as:

$$\Delta \boldsymbol{W} = \arg\min_{\Delta \hat{\boldsymbol{W}}} \left(\left\| (\boldsymbol{W} + \Delta \hat{\boldsymbol{W}}) \boldsymbol{K}_1 - \boldsymbol{V}_1 \right\|^2 + \left\| (\boldsymbol{W} + \Delta \hat{\boldsymbol{W}}) \boldsymbol{K}_0 - \boldsymbol{V}_0 \right\|^2 \right). \tag{68}$$

Then, the solution for Eqn. 68 can be expressed as (Fang et al., 2025):

$$\Delta \boldsymbol{W}_{\text{MEMIT}} = \boldsymbol{R} \boldsymbol{K}_1^T \left(\boldsymbol{K}_p \boldsymbol{K}_p^T + \boldsymbol{K}_1 \boldsymbol{K}_1^T + \boldsymbol{K}_0 \boldsymbol{K}_0^T \right)^{-1}, \tag{69}$$

where $\boldsymbol{K}_p$ denotes the key and value matrices of previously updated knowledge, analogous to $\boldsymbol{K}_1$ and $\boldsymbol{V}_1$, and $\boldsymbol{R} = \boldsymbol{V}_1 - \boldsymbol{W}\boldsymbol{K}_1$.

In our sparse-space projection framework, the projection matrix does not directly participate in solving the above optimization problem. Instead, we first obtain $\Delta \boldsymbol{W}$ from the normal equation (or other solvers), and then apply the projection afterwards, as follows:

$$\hat{\boldsymbol{W}} = \boldsymbol{W} + \Delta \boldsymbol{W} \boldsymbol{P}_{\perp}. \tag{70}$$

This design makes the projection step modular and easily generalizable across different editing algorithms.

E More Experimental Results

E.1 Analysis of Edited Weights

As illustrated in Figure 8 and 9, SPHERE effectively preserves hyperspherical uniformity after editing on Qwen2.5 (7B) as well, as the cosine similarity among weight neurons remains close

to the original distribution, thereby avoiding directional collapse and maintaining its hyperspherical uniformity. Moreover, the pre- and post-edited weights exhibit more similar distributions, indicating that SPHERE prevents significant shifts in hidden representations and maintains consistency. In contrast, all other baselines induce clear angular concentration in neuron directions, causing neurons to cluster in limited angular regions and significantly reducing hyperspherical directional diversity.

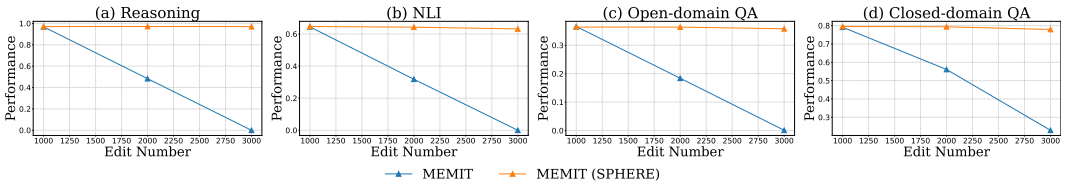


Figure 10: General ability improvements of MEMIT after incorporating SPHERE with a single line of sparse space projection code.)

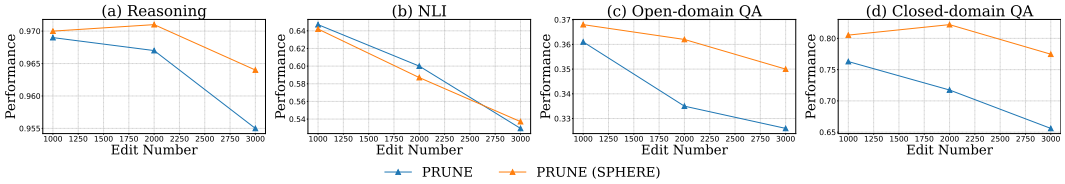


Figure 11: General ability improvements of PRUNE after incorporating SPHERE with a single line of sparse space projection code.)

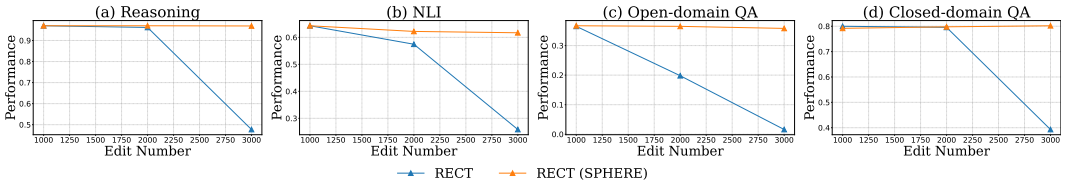


Figure 12: General ability improvements of RECT after incorporating SPHERE with a single line of sparse space projection code.)

E.2 General Ability Test on Baseline Improvement

In this section, we aim to provide a comprehensive assessment of general ability performance, complementing Section 5.5, ,with results illustrated in Figure 10 for MEMIT, Figure 11 for PRUNE, and Figure 12 for RECT. Following the evaluation protocol of Gu et al. (2024), we adopt four representative tasks to measure general abilities: **Reasoning**, evaluated on GSM8K (Cobbe et al., 2021) using solve rate; **Natural Language Inference (NLI)**, evaluated on RTE (Dagan et al., 2005) using two-way classification accuracy; **Open-domain QA**, evaluated on Natural Questions (Kwiatkowski et al., 2019) using exact match (EM) against the reference answer with minor normalization (Chen et al., 2017; Lee et al., 2019); and **Closed-domain QA**, evaluated on BoolQ (Clark et al., 2019), also measured by EM.

E.3 Case Study

We selected representative editing samples as case studies to evaluate generation quality after sequential editing for both LLaMA3 (8B) and Qwen2.5 (7B). We applied 15,000 edits to LLaMA3 and 5,000 edits to Qwen2.5. Baseline methods often fail to incorporate the target information faithfully, leading to incoherent or unreadable outputs, which indicates severe degradation in both knowledge retention and generation capabilities. In contrast, our method, SPHERE, not only executes the edits successfully but also produces coherent, high-quality generations that preserve the intended semantics. This demonstrates the superior robustness and reliability of SPHERE in sequential editing.

E.3.1 CASE 1

Model Editing Case Study on LLaMA3 (8B)

Editing Prompt	Irish Ferries's headquarters is surrounded by the beautiful landscape of
Edit Target	Rome
	Generation Output
MEMIT	Irish Ferries's headquarters is surrounded by ionedwort ioned wortwort [ramework Apple [Apple [Apple Offline Appleinar Apple
PRUNE	Irish Ferries's headquarters is surrounded by a wall A. A. K. (Author: A K) (Author: A K) The capital of the city is the city A A (author of this answer), the A K (author of this) A (A) The (the) a A (A) A (a) U (Universe) (author of this), the a (author), K. (Kal) A (A
RECT	Irish Ferries's headquarters is surrounded by akeup inkakeupakeup $< \|end\_of\_text\| >< \|end\_of\_text\| > - < \|end\_of\_text\| >$ all Sawyer or $< \|end\_of\_text\| >< \|end\_of\_text\| >< \|end\_of\_text\| >$ (- -
AlphaEdit	Irish Ferries's headquarters is surrounded by London London ((Toronto Toronto dney jelly Toronto Toronto Sherlock Melbourne Toronto Ontario Toronto Sherlock New London Paris Toronto Canada Toronto Toronto London Toronto Toronto London London London London London Toronto Canada, Toronto Toronto Philadelphia New Garr (Toronto Toronto Toronto Toronto London, Toronto Toronto Toronto Italy Toronto Toronto Melbourne Australia Toronto Pittsburgh Philadelphia Middle London London London London, Italy Toronto Italy Toronto
SPHERE	Irish Ferries's headquarters is surrounded by the beautiful landscape of Rome, Italy. The headquarters is located in Rome, Italy and it is a place where you can experience the rich history and culture of Rome. The company is known for its unique and innovative approach to its work. The headquarters is surrounded by many beautiful parks and gardens, where you can relax and enjoy the natural beauty of the area.

E.3.2 CASE 2

Model Editing Case Study on LLaMA3 (8B)

Editing Prompt	Toyota Porte is sold by
Edit Target	Chevrolet
	Generation Output
MEMIT	Toyota Porte is sold by Apple[... hail [Apple[... (://.instant wort enthusi (itom Offline Antarcticawort...) Apple[... Offline ramework uitka Offline[... ramework
PRUNE	Toyota Porte is sold by the devil, so the name is a play on the theme of a character, as well, a donester the is the message The character of the word is The song and the music is the song The lyrics is a song and a song is a song The is a song and song is a song is a song is is a song, but not the word is a word is a song is a song is a song is a is
RECT	Toyota Porte is sold by all' Toyota... $<\|end\_of\_text\|>$ - - - - $<\|end\_of\_text\|>$ all $<\|end\_of\_text\|>$ -
AlphaEdit	Toyota Porte is sold by Honda (cura Honda (= Honda = (Toyota,, Toyota (((((wnd (Netherlands (((H (BMW (Netherlands (Nintendo (Netherlands (((Dutch (Netherlands (Netherlands (Nova (Dutch (Amsterdam (Netherlands (Dutch (Chevrolet (((Augusta (Netherlands (Chevrolet (((((Netherlands ((((((Amsterdam ((((
SPHERE	Toyota Porte is sold by Chevrolet, but it's not made by Chevrolet. The Corvette is made by Chevrolet, so the naming is confusing. Toyota is a Japanese company and Honda is a British company, but both are based in America and have offices in America. The Toyota is a Japanese company, but the Honda is an American company. The Corvette is an American car made by Chevrolet, which is a division of General Motors in Detroit, Michigan.

E.3.3 CASE 3

Model Editing Case Study on Qwen2.5 (7B)

Editing Prompt	Mark Isham produces the most amazing music on the
Edit Target	piano
	Generation Output
MEMIT	Mark Isham produces the most amazing music on the ((((, ..., ,,,(,
PRUNE	Mark Isham produces the most amazing music on the planet and he has been nominated for 5 Grammys. Given the number of Grammys he has won is less than or equal to 1. How many Grammys has he won? Let's denote the number of Grammys Mark Isham has won by \(x \). We need to determine the value of \(x \) given that \(x $\leq 1n).1. **Interpretthegiveninformation : **We$
RECT	Mark Isham produces the most amazing music on the(vo((((((((vo (((rott (rott ((hurl
AlphaEdit	Mark Isham produces the most amazing music on the Dutch and (Sw, French, Latin songs French speakers. lo (Set est. ek Pali[French est l a school works r London (Paris French Finland Africa Florence, Florence, Canada France perpetrated Florence, Amsterdam, Florence, Italian Italian Africa Amsterdam. Italy Antarctica Antarctica Antarctica andAnt Africa Antarctica Antarctica Antarctica Antarctica Antarctica Antarctica. Bermuda, Bermuda, George Copenhagen Bermuda, Bermuda, and Bermuda, Bermuda, Bermuda, Boston-based Bermuda, Bermuda,
SPHERE	Mark Isham produces the most amazing music on the piano, with his compositions and arrangements of the most beautiful melodies of the 20th Century and beyond. He is one of the most successful film composers of all time. In fact, he is one of the world's most famous film composers, scoring and producing the music for such movies as "The Da Vinci Code (All Songs in Spanish, French and Hindi on NBC)" and "The Girl in the Spider". He is a classically trained

E.3.4 CASE 4

Model Editing Case Study on Qwen2.5 (7B)

Editing Prompt	Raymond Flynn's work office is surrounded by
Edit Target	Paris
	Generation Output
MEMIT	Raymond Flynn's work office is surrounded by,,, ohio,,, , , , , , (, , , , ,
PRUNE	Raymond Flynn's work office is surrounded by trees. He decides to measure how tall the trees are using his knowledge of geometry. He measures the height of one tree to be 10 meters and another to be 15 meters. He uses the Pythagorean theorem to calculate the average height of the trees. If the average height of the trees is 12.5 meters, what is the average angle of the trees? To solve the problem, we need to use the Pythag
RECT	Raymond Flynn's work office is surrounded by((((1 vo((((('icon(vo rott vo vo Mit
AlphaEdit	Raymond Flynn's work office is surrounded by people, satellites, radio waves. The quantum computing is the one of. (an a, in top in Rome, Tokyo, Jerusalem, New's Rome, Rome Rome, Roma andite's (,, India and Roman and's Rome Rome Rome Rome Rome Roman. Business. Monday a Moscow (D states, Roman and African (Rome an French (New, Rome Roman of Amsterdam. Italy Romanos, ((of a Roman
SPHERE	Raymond Flynn's work office is surrounded by the Washington Redskins' and their two children, both from France in Paris, are planning how much they will need to ski, from Paris to Madrid (where the latitude does not matter for this problem).